11-1-2009

Implementation of functional safety in a robotic manufacturing cell using iec 61508 standard and siemens technology

Darshana M. Kamtekar

IMPLEMENTATION OF FUNCTIONAL SAFETY IN A ROBOTIC MANUFACTURING CELL USING IEC 61508 STANDARD AND SIEMENS TECHNOLOGY

Darshana M. Kamtekar

B. E. (Electronics Engineering)

University of Mumbai, 2003

Thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in the Department of Industrial & Systems Engineering in the Kate Gleason College of Engineering of the Rochester Institute of Technology

November 2009

KATE GLEASON COLLEGE OF ENGINEERING

ROCHESTER INSTITUTE OF TECHNOLOGY

ROCHESTER, NEW YORK

CERTIFICATE OF APPROVAL

MASTER OF SCIENCE DEGREE THESIS

The M.S. Degree thesis of Darshana M. Kamtekar has been examined and approved by the thesis committee as satisfactory for the thesis requirement for the Master of Science degree.

Approved By:

_____	_____
Dr. Matthew Marshall	Dr. Benjamin Varela
_____	_____
Dr. S. Manian Ramkumar	Dr. Jacqueline Mozrall

I dedicate this Master's thesis

To my parents Manohar & Savita Kamtekar, who are the reason where I stand.

To my relatives Jagannath & Sunita Surve, who are there to support me in all circumstances.

To my sister and brother, Gauri & Krushnaraj, who are ever present for motivating me.

To all my friends and colleagues who made life an exciting adventure.

ACKNOWLEDGEMENTS

I am indebted to my advisor, Dr. Benjamin Varela, for his guidance and support throughout the course of the research work. I would like to thank him for his contribution and encouragement, which have seen me through the successful completion of this thesis. I offer my sincerest gratitude to Dr. S. Manian Ramkumar for his proactive contribution and guidance as a thesis committee member. His innovative approaches and ideas made the research work enjoyable.

I would like to thank Dr. Matthew Marshall for his association with this thesis as an advisor. His suggestions and advice have been invaluable. I would also like to take this opportunity to thank Dr. Jacqueline Mozrall, head of the Industrial and Systems Engineering department, for all the support that helped me accomplish all the thesis requirements. I would also like to thank all the faculty and staff members of the Industrial and Systems Engineering Department for their direct or indirect support during my tenure of thesis study.

I would like to thank Siemens Automation for their contribution to this work in terms of financial aid, product supplies and technical assistance during the execution of this thesis.

Finally, I would like to thank all my friends for their undue help in good and bad times throughout my tenure of thesis study.

ABSTRACT

The past 50 years have seen a staggering amount of change in the technology and the business of process automation. The programmable logic controller (PLC) based control and monitoring system is a proven technology used to not only control processes but also to perform safety functions for processes in many industrial applications. There are many opportunities for improvements in any process or manufacturing system. One of the opportunities is achieving accurate safety function for measurement and process control to prevent human injury or death. The programmable electronic systems (PES) such as PLC systems are increasingly being used to perform safety functions as an integral part of the process or plant control system. A Robotic Manufacturing Cell is an example of a PES system and is used as an experimental setup for this work. The IEC 61508 standard defines various phases involved in the overall safety lifecycle for the PES system. This thesis study concentrates on such phases that include safety analysis methods, selection of an appropriate safety control system, implementation of safety as per the standard and safety validation. In this study four test cases are selected to perform safety analysis and implementation. It is verified how the conventional safety analysis method (FMEA) can be used to estimate the risk associated with each test case. As recommended by IEC 61508, a Risk-Graph method is used to calculate the Safety Integrity Level (SIL) requirement for each test case. A number of factors are required to be considered for selecting the appropriate safety control system architecture. After studying these factors and the safety analysis results, the Siemens safety PLC-based control system with SIL 3 configuration is selected for this application. IEC 61508 also recommends implementation of independent control systems for normal operation

and safety. This study demonstrates how two independent PLC based control systems, one for normal operations and other for safety-related functions, are implemented to offer the most effective solution for this application. This is achieved by using PLCs from two different manufacturers, a non-safety PLC for normal operations and a Siemens safety PLC for safety-related functions. This study focuses on Machine Safety, and it can be used as a guideline for implementation of functional safety in real-life manufacturing environment.

Index

List of Tables

List of Figures

Chapter 1

Introduction

1. Introduction

The past 50 years have seen a staggering amount of change in the technology and the business of process automation. It is admitted that automation enables sophisticated process control and handling of unwanted errors without human interference[1]. The programmable logic controller (PLC) based control and monitoring system is a proven technology used to not only control processes but also to perform safety functions for processes in many industrial applications. The PLC based distributed control system enables the engineer to gather information from all processes across the plant and control the operation from a central control room.

There are many opportunities for improvements in any process or manufacturing system. One of the opportunities is achieving accurate safety function for measurement and process control to prevent human injury or death. The programmable electronic systems (PES) such as PLC systems are increasingly being used to perform safety functions as an integral part of the process or plant control system. New technological and industrial development has led to a demand for PLC based system to perform safety-related functions in many applications.

Safety is defined by many researchers in various ways. According to K. C. Shen's definition [1,2], "safety of a system is the probability that, when operating and/or residing under stated conditions, the system will not be injured significantly for a specified interval of time." Safety in a process plant can influence the design of a process control system in such a manner that hazard to machinery and humans can be avoided. Traditional plant designs try to

reduce the risk by adding personal protective equipment and by applying standard operating procedures (SOPs) [3]. In a conventional environment, safety engineers are assigned to prove that an existing design is safe. If a safety engineer discovers significant safety problems late in the design process, correcting them can be very expensive and time consuming. Instead, a safety engineer can be involved at the early stages of the design to finalize the safety specifications.

The Occupational Safety and Health Administration (OSHA) is an agency of the United States Department of Labor. Its mission is to prevent work-related injuries, illnesses and occupational fatality by issuing and enforcing rules called standards for workplace safety and health[4]. In case of any regulatory violation, OSHA enforces penalties on the responsible authorities. Due to this, safety in the workplace has gained importance for every manufacturing industry.

"Safety integrated" [5] is a term widely used by Siemens Automation and Drive (A&D) group to promote their safety concept. This concept allows engineers to use standard as well as safety components to create safety-integrated, cost-effective solutions depending on the Safety Integrity Level (SIL) requirement. Siemens, being a pioneer in automation and control industries, has developed various products for machine and process control with proven technology. Having provided a variety of standard products known for their reliable operation, Siemens has developed and manufactured various components required in automation with safety incorporated in it. The safety-integrated product series include SIRIUS, SIGUARD, SIMATIC and SINUMERIK/SIMODRIVE products that are

configured to provide maximum protection against functional faults. The SIMATIC S7 Distributed Safety is a safety related programmable system certified by TUV SUD (German Technical Inspectorate, SOUTH). This means that it is suitable for use in safety-related applications with high potential hazards and risks such as production systems, machinery construction, process technology and offshore processes. The certification is aligned to IEC-61508.

IEC 61508 is a globally accepted standard and is used to implement functional safety for PES systems. This standard explains an overall safety lifecycle. This document can be used as a guideline for the design and implementation of a safety-related system. This thesis demonstrates how the IEC 61508 is used to develop a safety-related system by using Siemens safety products.

Chapter 2

Background

2. Background

2.1 Overview

It was found that the safety requirements for each process industry vary depending on their application, breadth and complexity. Any process plant consists of a large number of processes and each process is operated by using a number of mechanical, electrical and pneumatic components. The safety requirements are different for design and implementation of each component. The safety experts follow different standards to design and implement each of the above components. Automation is an essential component required to operate a process. In the academic literature, many topics related to safety control systems are discussed. These discussions include conventional methods of safety analysis, different types of failures, elements of safety, various configurations of control systems and IEC 61508 safety standard and its application. A PLC based control system for safety is an emerging area and there are not many articles published in the academic literature yet.

2.2 Conventional Methods of Safety Analysis and their Application

Methods of safety analysis can be applied during the early designing stages of process automation. There are methodologies and procedures available for analyzing the hazards in the process industry that can be applied to other industries. A. Toola [1] has suggested that the conventional safety analysis methods can be applied to the safety design of a PLC-based control system. The author has reviewed many case studies on safety analysis methods used

for automation. The author has also listed scope, principles, advantages and shortcomings of various safety analysis methods such as:

1) Hazard and operability study (HAZOP)
2) Action error analysis (AEA)
3) Fault tree analysis (FTA)
4) Event tree analysis (ETA)
5) Failure mode and effect analysis (FMEA)
6) Reliability assessments

Many researchers have studied and analyzed each method with respect to an industrial application and have listed its advantages and disadvantages. Few researchers have also shown that two or more methods can be used in combination for specific application. All these methods are summarized based on their scope and principles in Appendix 1[1,6,7,8,9,10].

2.2.1 Checklist

Checklist[11] is one of oldest and simplest method used for hazard identification. A checklist is a list of questions about system organization, operation, maintenance and other areas of concerns. It can also help to determine appropriate actions required for hazard control. The implementation of checklist assures that various requirements are fulfilled and nothing is neglected or overlooked. Checklist is primarily based on the analyst's prior experience. It can also be implemented based on codes and standards. The checklist is required to be maintained during the life of the project and updated after every major and substantial

modification. Although checklist requires relatively trained and experienced people, relatively untrained person can also use it effectively with adequate resources.

A detailed qualitative assessment can be carried out by using number of checklists. B. K. Daniels and R. I. Wright [10] have suggested few main headings for various checklists, such as Safety related functions, Operator interface, Plant interface, Physical environment, Maintenance and modifications. Each heading is followed by their subsidiary sections and questions. This method can be an extensive amount of work to cover all forms of failures.

The main disadvantages of this technique [11,12] are as follows;

- It takes a long time to develop a checklist. It yields qualitative results but no insight into the system. It just provides the status of each item in terms of 'Yes' or 'No.'
- A checklist can focus on only single item at a time. It cannot find hazards which are a result of interaction among different equipment.
- Because it is required to be prepared by an experienced person, there is always a significant probability that some critical failures are being neglected.
- It is not possible to identify the causes of hazards such as type of equipment operation, severity of operating conditions and any mis-operation.

Due to above listed limitations, this method is not recommended for details risk assessment.

2.2.2 What-If Analysis

The What-If method [12] involves asking a series of questions beginning with What If (not necessarily start with “What if”) as a means of identifying hazards. Apart from checklist, it is the oldest method of hazard identification and is still popular. What if analysis is performed with questions such as:

- What if the pipe leaks?
- What if the flow controller fails?

This method essentially involves a review of the earlier design by a team using questions of this type, often using a checklist. The advantages of this technique are:

- No specialized technique or computational tool is required.
- Once the questions have been developed they can be used throughout the life of the project.
- It provides a simple tabular summary.

The major disadvantages are:

- It is recommended to have a team of experts to perform the study. As a result it has disadvantage in terms of expertise availability and cost.
- The heavy reliance on the experience and intuition of the study team implies that any limitations in this aspect of study can make the entire study useless.
- It is not as systematic as HAZOP and FMEA.
- It gives only qualitative results with no numerical prioritization.

Due to these disadvantages, this technique can be used only when HAZOP and FMEA are not applicable or the cost of study is the main consideration.

2.2.3 Fault Tree Analysis (FTA)

Fault tree analysis[11] is the basis for a structured approach to failure analysis. It is an analytical tool that uses deductive reasoning to determine the occurrence of an undesired event. The technique begins with a top event that would normally be a hazardous event. Then the fault tree analysis is used to identify various single point failures, combinations of these failures and operating circumstances which could cause that event. The completed fault tree is a logical representation of all the combinations of basic event which cause the top event. Each basic event is considered as a Boolean variable, and the logic for top event can be represented as a Boolean expression[10]. By manipulating this expression, single point failures and minimum cut-sets are determined to identify various failure modes in PES.

FTA provides quantitative information about failure modes and consequences. FTA has the following advantages[11,12]:

- It allows the analyst to concentrate on one particular system failure at a time.
- It makes it easy to identify single point failures.
- It provides a graphical format which enables the analyst to visualize the hazard and its causes.
- It is used for some control systems to incorporate the effects of feedback.
- It can also handle multiple failures.

Software fault tree[1] (SFTA) attempts to verify that the program will not, in any environment, allow a particular unsafe output to occur.

FTA is a well-accepted technique. Its main disadvantages are that developed fault trees can be very large and difficult to relate to the system and its operations. The results can be difficult to quantify. Its accuracy relies on the ability of the analyst to deduce what can cause an event.

2.2.4 Failure Mode and Effect analysis (FMEA)

FMEA is a systematic examination of the system to determine the effect of each mode of the failure of each part of the system. In this analysis, individual components such as pumps, valves and vessels are examined to identify the likely failures which could have undesired effects on the system operation[11]. FMEA is a qualitative inductive method and is easy to apply[10]. FMEA is executed by preparing the list of expected failure modes with respect to the use of the system, the elements involved, the mode of operation, the operation specifications, the time constraints and the environmental conditions. It can be applied at any level of breakdown of the system, e.g., sub-system, module or components. It has been recommended for use as a hazard identification technique mainly for systems dealing with low/ moderately hazardous operations and the one which can not support the expensive and time-consuming HAZOP[12]. When applied to PES systems, it is usually applied at functional block levels. In this approach, effects of each mode of failure of field sensors, actuators, operator interfaces, processors, I/O modules and communication interfaces are considered[13]. FMEA is good at identifying potentially hazardous single failures but normally does not consider multiple and simultaneous failures.

2.2.5 Hazard and Operability Study (HAZOP)

HAZOP was developed by Imperial Chemical Industries (ICI) in the UK. It is a simple yet structured method for hazard identification and assessment. The basic principle of a HAZOP study is that normal and standard conditions are safe. The hazard occurs only when there is a deviation from the normal condition. The UK ministry of defense awarded a contract to Cambridge Consultant Limited working with Arthur D. Little and Redmill Consultancy to prepare a guideline on the application of the HAZOP to PES[14].

In a typical HAZOP study, design and operation documents such as piping and instrumentation diagrams (P&IDs), process flow diagrams (PFDs), material flow diagrams and operating manuals are examined systematically by a group of experts for identifying all possible deviations [15]. Once a HAZOP has identified all deviations, it searches for the cause of the deviation and tries to deduce the consequences of the deviation. To cover all possible malfunctions in the system the HAZOP team members are guided with a set of guide words for generating the process variable deviations. A list of guide words [8,12,14] and their definitions with respect to chemical industry and PES is given in Table 2.1.

It can be observed from the above description of methods that no safety analysis method covers all aspects of safety design. Each method has its own targets and is applicable to specific problems. However, it is possible to find a combination of methods which is optimal for each design problem.

Guide Word	Standard interpretation for chemical industry	Example interpretation for PES
None	No part of the intention is achieved	No data or control signal passed
More	Quantitative increase	Data is passed at higher rate than intended or more data is passed
Less	Quantitative decrease	Not used here because it is already covered by 'part of'
As well as	All design intent achieved but with additional results	Not used here because it is already covered by 'more'
Part of	Only some of the intention is achieved	The data or control signals are incomplete
Reverse	Covers reverse flow in pipes and reverse chemical reactions	Normally not relevant
Other than	A result other than the original intention is achieved	The data or control signals are complete but incorrect
Early	Not used	The signal arrives too early with reference to clock time
Late	Not used	The signal arrives too late with reference to clock time
Before	Not used	The signal arrives earlier than intended within a sequence
After	Not used	The signal arrives later than intended within a sequence

Table 2.1: Guide word interpretation with respect to the chemical industry and a PES system.

F. Redmill et al.[14] have suggested a combination of HAZOP and FMEA in hazard analysis. The two methods are complementary. When HAZOP is being carried out, there is often some

element of FMEA included in it, but if the difference between the two methods is understood, their effectiveness can be optimized. These methods can be briefly distinguished as follows;

- A HAZOP is a team exercise, while FMEA can be performed by an individual.
- HAZOP is used to identify both the causes and consequences of hazards while FMEA examines only the consequences of failures of each component.
- In HAZOP, once the deviation from desired intent is found, the study proceeds further to identify possible causes and likely consequences of the deviation, whereas in FMEA, once the possible components failures are identified, a study proceeds further to determine the likely consequences on the system as a whole.

When the hazard is a result of the deviation from design intent of either a component or interaction between components, neither the HAZOP nor FMEA alone can cover all possible hazards. However, when HAZOP provides a possible deviation from design intent as an interaction between components and possible cause as a failure of one of the components, FMEA can be applied further to investigate the possible causes of failure of that component. In another way, when the component is a subsystem and FMEA has recognized a possible failure mode, HAZOP can be applied on more detailed design representation to understand the interaction within the components of a subsystem. Therefore, it is possible to improve the efficiency of hazard analysis by carrying out first HAZOP and then FMEA or vice versa.

2.3 Elements of Failure

All of the above mentioned safety analysis methods consider three basic elements in safety assessment. These are causes of failure, probability of failure and consequences of failure. The causes of failure can be divided into two categories: the systematic failures and stochastic failures[1]. Examples of systematic failures are software errors, errors in design, specifications, construction, operation and maintenance. Examples of stochastic failures are failures due to the aging of mechanical components and random failure of electronic components. Probability of failure is the likelihood that the system will fail. It also defines the rate at which the equipment may fail during its lifecycle. The likelihood is classified as high, medium or low rate of occurrence. This is often determined based on company operating experience or industry-wide operation history [6]. The consequences are defined as the effects of failure on human life, property or the environment.

Angela Summers[6] has indicated in her study that once the HAZOP safety analysis is completed, risk associated with the severity and likelihood should be understood. The event severity is determined based on its anticipated consequences and impacts. This can include;

1. On-site Consequences
    - worker injury or death
    - equipment damage
2. Off-site Consequences
    - community exposure, including injury and death
    - property damage

3. Environmental Impact

    - emission of hazardous chemicals
    - contamination of air, soil and water supplies
    - damage to environmentally sensitive areas

The occurrence of a failure does not cause the hazard. Typically, a specific state of the process and/ or a combination of failures is needed for a hazard to occur. Multiple failures may occur simultaneously, and it is difficult to predict the behavior of a control system in such a scenario. The automation engineer can affect the probability and severity of accidents by applying hazard analysis methods. Each hazard analysis method is applicable for different categories of hazards. Sometimes it is necessary to use two or more methods simultaneously to cover all possible hazards. Risk reduction methods can be applied at various stages of failure as presented in figure 2.1 [1];

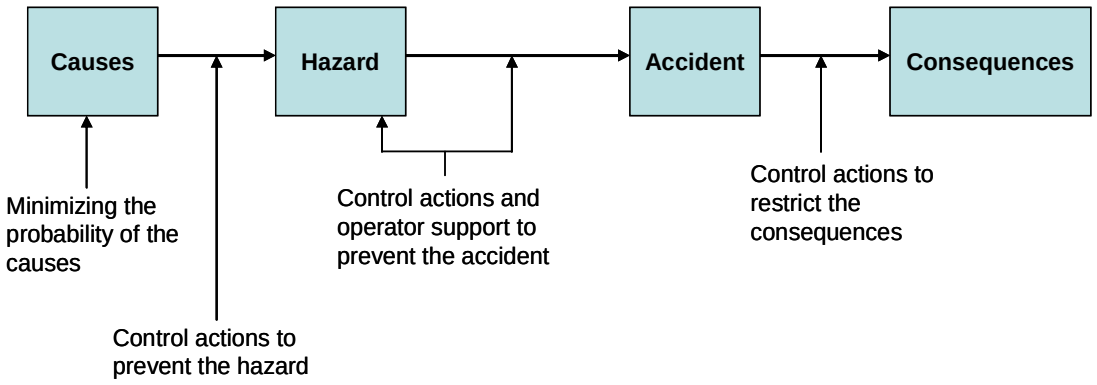


Figure 2.1: Opportunities for risk reduction.

.

The above figure represents various stages through which a failure goes before it causes the hazardous consequences. There are many opportunities for an engineer to reduce the risk. The potential causes of an accident which exist in automation can be eliminated, or their

probability can be minimized. A feedback from all stages can be fed back to the system to minimize the probability of the causes. Systematic failures can be minimized by designing procedures, by management actions, by having skilled and experienced designers and by having thorough testing. Stochastic failures can be minimized by using adequate quality components, by aging procedures and by testing procedures for redundant components. Redundancy at component and/ or architectural level can be provided. It is also possible to prevent the unsafe state by designing countermeasures for hazards by which the system can be brought back to safe state. This can be done by designing alarms, using automatic control actions, designing fail-safe systems, and interlock and trip systems and emergency shutdown systems. It may be possible to control unsafe state or the consequences of the hazard and to minimize them so to keep the process in a state which, though an unsafe state, still prevents any more harm from occurring. This can be done by means of safety protection equipment or by keeping people and material out of hazardous area [1].

A. Toola has presented various cases on accidents caused by the automation system. He reviewed literature on accidents by various researchers [1]. All of the above cases include the safety analysis on automation failures. These researchers have mostly used hazard and operability studies, fault tree analysis and failure mode effect analysis methods for analyzing effects of automation failures. It is also identified that the safety of automated control systems should include safety-of-application software. Software fault tree analysis (SFTA) is an extended FTA which can be used for application software. In this method, the “TOP” event is critical software fault, and the software is studied backward through the program to the software input. A. Toola has indicated in his safety analysis study that some methods do

not study multiple failure situations systematically. The simultaneous occurrence of failures of different types, for example, a human error in connection with a component failure, are difficult to study with a single safety analysis method. An automation engineer is required to use two or more safety analysis methods to cover various types and combination of failures. Safety analysis provides a practical way for automation engineers to discuss systematically with operators and process engineers the intended and unintended functions and states of a process.

According to Toola's accident studies [1], an important feature in automation safety is the information on process states provided to the operator. The safety analysis methods do not consider this aspect explicitly, but they provide information on the process disturbances which might lead to the unsafe states during operation. Only Action Error Analysis (AEA) has a comment on it, by asking in what way the operator notices his or her mistake. This information can be used to design the operator interface such as Human machine interface (HMI) application to keep the operator updated with latest state of the process. The HMI with real-time status updates is provided as an integral part of the automated control system. Figure 2.2 shows the various stages of failure caused by unintended event in human activity, in the technical equipment or in the environment [1].

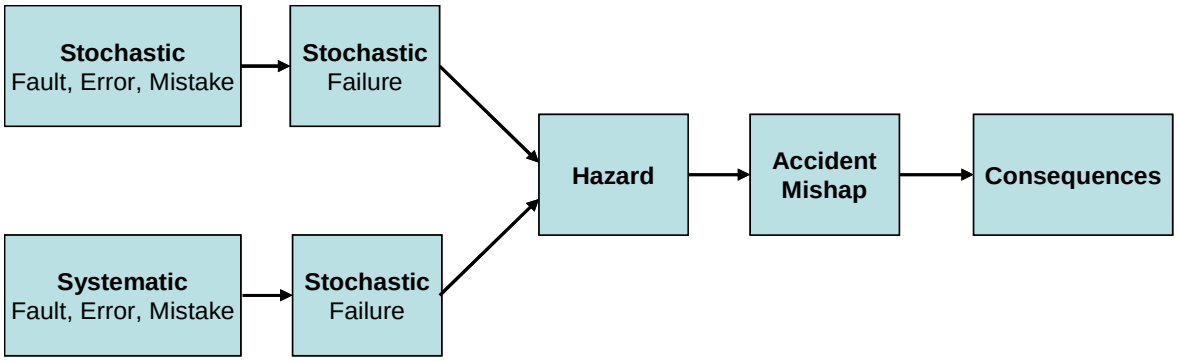


Figure 2.2: Various stages of failure caused by an unintended event.

2.4 Different Configurations of Automated Control Systems

The safety of a manufacturing process is based on the process design itself. The design of the process should define the number of possible unsafe states and their probability of occurrence. Many safety features can be implemented with the aid of automated control system. The objective of the automated control system is to keep the process in a safe state to prevent an accident and/ or to transfer the process back to a safe state in case of failure. The automated control system can be designed by using solid-state relay logic, embedded controller, microprocessor, PC based control cards or PLC. The PLC-based control system designed for safety has a fail-safe operation mode [7,16]. The purpose of a fail-safe system is to bring the controlled process to a pre-defined "safe state" in case of failure. The operation of the fail-safe system can be triggered by detecting quality concerns, environmental constraints, hazardous conditions or other unacceptable operational parameters. The PLC based control system can also be configured for fault tolerant operations. The fault-tolerant systems [16,17] have internal redundant components and integral logic for identifying and bypassing faults without affecting the output. Redundant systems have individually specified secondary components. A hardwired or software means are provided in PLC system for

detecting failure of the primary device. When the PLC system detects the failure of a primary device, it switches its control to the secondary device automatically.

Redundancy improves the availability of the system. The PLC can be configured in various ways to achieve redundancy at various levels. Some examples include Hot-Standby and Warm-Standby configurations for redundancy at CPU levels, redundancy at power supplies, voting logics (1oo2, 2oo3) for redundancy at I/O and field device levels, redundant industrial Ethernet network for redundancy at communication level and Triple Modular Redundant (TMR) systems for redundancy at all levels. Hot standby PLC system contains two CPUs connected in parallel configuration. A hardware module is provided to perform the switchover in case of failure of the primary CPU. The switchover time generally takes approximately 13-48 milliseconds. Warm-standby PLC system contains two CPUs connected in parallel configuration. A software program is loaded in the CPU to detect the failure and to perform the switchover. When the failure of the primary CPU is detected by a software code, entire data is transferred into the secondary CPU memory and secondary CPU takes the control of the system. The switchover time in this case is approximately 500-1000 milliseconds and depends on the amount of data needed to be transferred. Redundant I/Os can be achieved in two ways. They can be done by using redundant sensors or actuators in the field or by using redundant sensors/ actuators with redundant I/O modules in the PLC system. Voting logic can be implemented to acquire information from these redundant sensors. This logic can be executed by using external voting electronic hardware or by using PLC I/O modules with built-in setting such as 1oo2 or 2oo3. In case of 1oo2 (one out of two channels) selection, the discrepancy period is monitored between two channels. If the status

of both channels is switched to the same state within the discrepancy period, the resulting status is read and stored in the CPU memory. If the status of both the channels is switched to same state beyond the discrepancy period, the fault is generated. The discrepancy period is decided based on internal circuit response time and the field device response time. Similarly, 2oo3 (two out of three channels) is implemented. Triple modular redundant (TMR), as the name suggests, has three redundant hardware modules. The switchover is performed similar to the Hot-standby system but has an additional level of redundancy. The goal of a TMR system is to provide fail-safe control in a fault tolerant mode. This allows the system to continue while any single fault is detected, diagnosed, isolated and repaired before the second fault can occur [16]. The availability of the TMR system is always more than the Hot-standby or Warm-standby PLC system. TMR systems are being used in highly critical applications such as nuclear reactor control. Redundant networks are implemented by using two Ethernet cards with each PLC. Both cards are configured for two independent networks. In case of one ethernet card failure, redundant card takes over the network control. HMIs are configured to automatically switch over to the available network addresses so that the real-time information is always available for the operator. “Managed Industrial Ethernet switches” are used for forming Redundant and/or Ring Ethernet network. Moxa technologies provide managed switches which can recover the network from failure within 20 milliseconds.

Reliability of the control system is often confused with safety [18]. Reliability is a measure of the “up-time” or availability of the system. It is normally measured with the Mean Time Between Failure (MTBF) and Mean Time To Repair (MTTR). MTBF is a statistical measure

of probability of failure. MTBF numbers represent a statistical approximation of how long a set of devices should last before failure. It does not mean that the device is tested for long time interval. MTBF numbers are generally created by estimating the MTBF of individual components and by past experience with similar products. Generally, manufacturers of industrial components or equipment provide the MTBF information for the user. Some methods are available that can be used to reduce the system down-time by increasing the MTBF value for example, conformal coating for printed circuit boards (PCBs) used in a system. A standard MIL-I-46058C specifies the conformalcoating requirement for Electrical/ Electronic Printed Circuit Assemblies. The conformal-coated modules last longer compared to standard modules in corrosive environment. If a process plant is situated near coastal area, this standard can be used as a design specification for PLC control system. A redundancy is one of the design approaches that increases the availability as discussed above. Another approach is to provide partial system functioning. In this case, operations that are critical to the production are still running, even if other processes are shut down due to the fault. MTBF and MTTR values are used to measure the availability of the system according to the following formula [16].

$$A = MTBF/ (MTBF + MTTR) \quad \text{----------(2.1)}$$

Mean Time To Repair (MTTR) is a composite of other terms such as Mean Time To Diagnose (MTTD) and the Mean Repair Time (MRT). If the system is designed for fail-safe control, then the MTTD is zero, as it causes the system shutdown by bringing the entire

system in a pre-defined safe state immediately after fault is detected [16]. Figure 2.3 shows the effect of safety on the % MTTR of the system

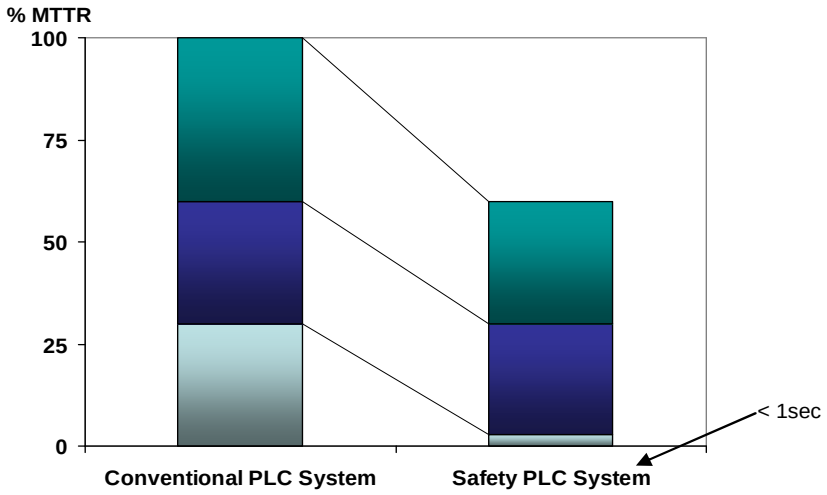


Figure 2.3: The effect of safety on the % MTTR of the system[16].

2.5 International Standards for Functional Safety of the Control System and Their Applications

According to a functional safety application study done by H. Kanamaru et. al, there are two methods of decreasing the risk in process automation [19]. One is intrinsic safety and the other is functional safety. In intrinsic safety, the workers and machines are separated physically by a guard or a barrier. In functional safety, the safety control system stops the process when it detects anyone intruding into hazardous area or when it diagnoses any fault. The functional safety system needs a combination of safety input devices, safety circuit and safety output devices. Functional safety is a part of overall safety that depends on a system or equipment operating correctly in response to its input.

Currently, there are many standards available for the planning, construction and operation of Safety Instrumented System (SIS). In the European Union, manufacturers of such systems could refer to safety standards such as DIN/VDE 19250, DIN/VDE 19251, DIN/VDE 801, EN 298 and EN 954. The design of control system for safety functions can be described using these standards. Since many countries have different standards which are used for different applications, a globally applicable IEC 61508 [20,21,22] basic standard was developed and adopted. The International Electrotechnical Commission (IEC) is the world's leading organization that prepares and publishes International Standards for all electrical, electronic and related technologies - collectively known as "electrotechnology." The IEC has issued standards for electricity and electronics, supporting safety and performance, the environment, electrical energy efficiency and renewable energies. The IEC also manages conformity assessment systems that certify that equipment, systems or components conform to its International Standards.

IEC 61508 standard is concerned with functional safety achieved by safety-related systems that are primarily implemented in electrical and/ or electronic and/ or programmable electronic (E/ E/ PE) technologies, i.e., E/ E/ PE safety-related systems.

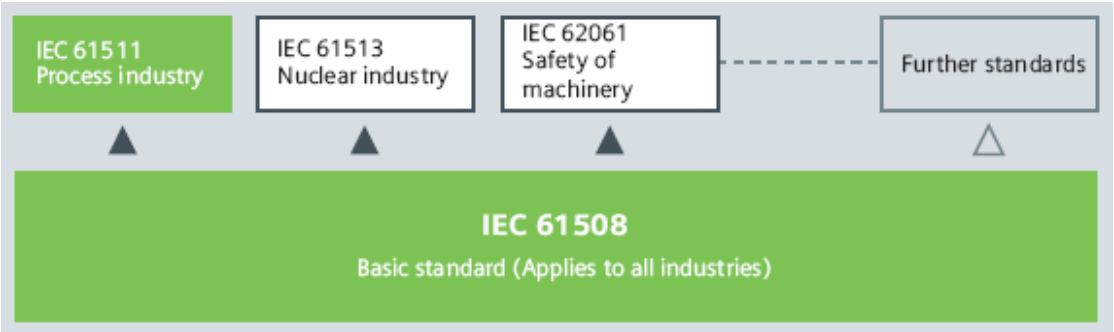


Figure 2.4: Standards used for functional safety[21].

As shown in the figure 2.4, the IEC 61508 standard is very generic and applies to all safety-related systems irrespective of the application sectors such as process industries, manufacturing industries, transportation, medical etc. IEC 61508 is also used as a foundation to develop safety standards applicable to other industrial sectors. For example, IEC 61511 is applicable to the process industry, IEC 61513 is applicable to the nuclear industry and IEC 62061 is applicable to machine safety.

Some key features covered in IEC 61508 standard [23,24] for a safety assessment are as follows:

1. Uses of a risk-based approach to determine safety integrity requirements of E/ E/ PE safety-related systems and including a number of examples of how this can be done.
2. Uses of an overall safety lifecycle model as the technical framework for the activities necessary for ensuring functional safety is achieved by the E/ E/ PE safety-related systems.
3. Uses of the safety life-cycle activities from initial concept through hazard analysis and risk assessment, development of the safety requirements, specifications, design and implementation, operation and maintenance and modification, to final decommissioning and/ or disposal.
4. Includes the systems and sub-systems designed to perform safety functions and failure modes for each component included in these systems.
5. Specifies requirements for both preventing failures and controlling failures.
6. Specifies the techniques and measures that are necessary to achieve required safety integrity.

IEC 61508 standard has seven-part framework (Appendix 2) that specifies the procedural steps to be performed during the design of the safety system. According to the IEC 61508, the first step in determining requirements for E/ E/ PE safety-related system is risk assessment. The risk assessment methods mentioned earlier are classified into two major categories. One is quantitative method and the other is qualitative method [21].

Quantitative methods are often used when there is limited historical information available about the process. These methods require a thorough understanding of the potential causes of failure and its estimated probability. The probability of failure is the rate at which a hazardous event can occur without existing protective measures multiplied by the effect of the event. The probability of failure can be estimated by analyzing the rate of failure in similar situations, by referring to historical records or by using analytical methods. The fault tree analysis (FTA) is an example of a quantitative method.

In qualitative method, a risk can be calculated based on the extent of damage (C) and the frequency of occurrence of the damage (H). The frequency of occurrence of the damage is a function of three elements:

a. The exposure to hazardous area (A)
b. The possibility of avoiding the hazard (G)
c. The probability of the unwanted event without any protective equipment (W).

Figures 2.5 and 2.6 represent how the quantitative method is used to determine the risk according to the IEC 61508 standard. IEC 61508 standard includes the Risk-Graph and associated safety integrity levels [20,21,25].

Extent of damage	
C_a	Light injury of a person, small environmental damage
C_b	Severe injury or death of a person
C_c	Death of several persons
C_d	Death of very many persons
Duration of stay of a person in the dangerous area	
A_a	Seldom to frequent
A_b	Frequent to permanent
Aversion of danger	
G_a	Possible under certain conditions
G_b	Hardly possible
Probability of occurrence	
W_1	Very low
W_2	Low
W_3	Relatively high

Figure 2.5: Various levels of A, G, W and C considered to determine the SIL[21].

This standard also describes four levels of safety integrity for the safety-related equipment. Safety integrity level (SIL 1) is the lowest level of safety integrity and SIL 4 is the highest that can be achieved with E/ E/ PE system. For example, the following table 2.2 shows how to determine the SIL level based on different values of four variables (C, A, G and W).

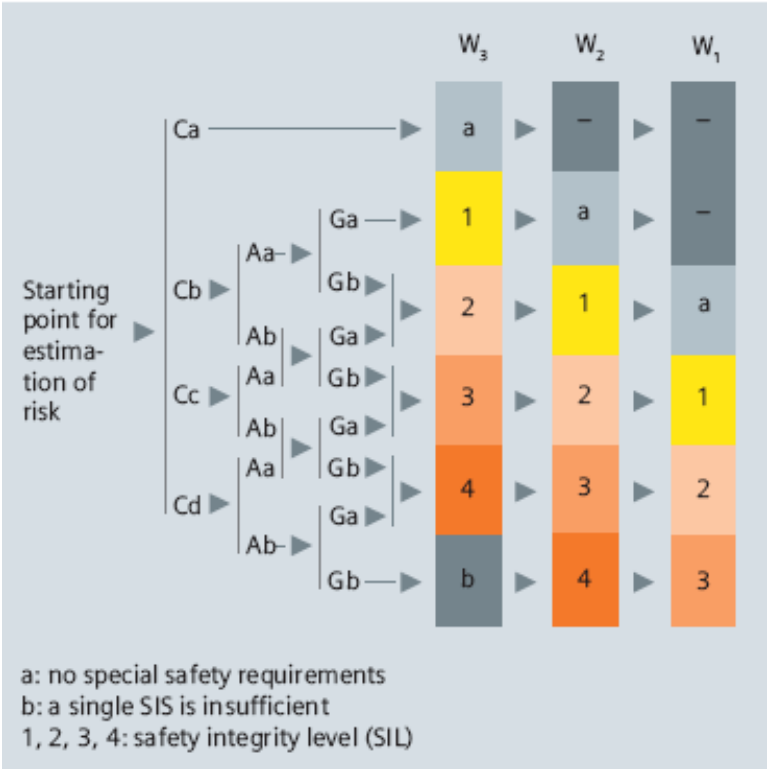


Figure 2.6: Determination of SIL according to the "qualitative method" [21].

Variables	**Value**	**Level**
Extent of damage, C	C_b	Severe injury or death of a person
Duration of stay of a person in the dangerous area, A	A_b	Frequent to permanent
Aversion of danger, G	G_b	Hardly possible
Probability of occurrence, W	W_3	Relatively high
SIL level required	**SIL 3**	

Table 2.2: An example of SIL calculation using a risk graph.

An automated control system has to be configured to achieve the required safety integrity level. The instrumented system (SIS) is examined in its entirety for SIL levels. To achieve required SIL, entire SIS has to be examined. All components of SIS, from sensors to

actuators, are investigated for the failure probabilities (PFDs) for calculating final SIL. In actual practice, the physical connection and/or bus communication is also considered in the final SIL calculation.

The safety integrity levels and associated range of probability of failure on demand is specified in the IEC 61508 standard [6,7,20,21,23].

Safety Integrity Level (SIL)	**Probability of Failure on Demand**
SIL 4	$>= 10^{-5}$ to $> 10^{-4}$
SIL 3	$>= 10^{-4}$ to $> 10^{-3}$
SIL 2	$>= 10^{-3}$ to $> 10^{-2}$
SIL 1	$>= 10^{-2}$ to $> 10^{-1}$

Table 2.3: SIL levels and associated range of PFD

In earlier days, the conventional PLCs were used with external relay logic to achieve the required safety function. This requires a lot of engineering, verifications, testing, validation and commissioning efforts. As the safety requirements become more critical, the system becomes more complex and so does the engineering and commissioning efforts. The study done by T. A. Walczak shows that field devices such as sensors and actuators outside of the PLC based control system hardware constitute the majority of failures within the overall control system architecture [16].

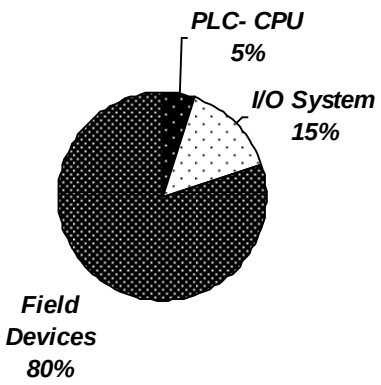


Figure 2.7: Control System Failure Topology by T. A. Walczak[16].

The implementation of functional safety in an automated control system ensures that the failures at sensor and actuator levels are detected and diagnosed within the safety time limit to achieve the overall safety of the system.

2.6 Introduction to Safety PLC, I/O and Safety Sensors

The manufacturers of the programmable logic controller systems have realized the importance of safety in early 1990's. Manufacturers like Rockwell, Siemens and GE-Fanuc have designed and developed PLC systems which meet the safety-related functional requirements as per the IEC 61508 standard. This new series of product is called safety-PLC[17,18,19,26,27]. A safety PLC provides high reliability and high safety via special electronics, special software and pre-engineered redundancy. A safety PLC has I/O circuits that are designed to be fail-safe with built-in diagnostics and read-back capabilities. The CPU of the safety PLC has built-in diagnostics for memory, CPU operation, watchdog timer and

all communication systems. The extensive diagnostics make it possible for the automation engineer to troubleshoot highly complicated installations. It is also possible to design the countermeasure so that the entire process can be completely or partially shut down in case of failure. Few safety PLCs also offer muting mechanism which can be used to switch off the safety features conveniently during maintenance or troubleshooting activities[28]. Built-in redundant architecture increases the availability of the system ensuring that it is operating in safe state. The feature of diagnosing fault to its last connection level makes it easy for the plant engineer to maintain and repair the widely spread distributed control system.

Safety PLCs are supplied with engineering tools for safety programming. The safety PLCs can be programmed by using five languages certified by IEC 1131-3 application programming standard. Manufacturers also provide library of safety-certified standard program blocks for ready to use. These software blocks are capable of monitoring and updating safety input and output at very high speed in the CPU memory. Safety PLCs can be configured with standard PLCs in one control system. In industrial practice, the normal process control is performed by standard PLC and safety PLC is used to perform highly critical functions. Both PLCs can interact with each other on safety network and thus reduce the ambiguity in process control decisions. With specially developed hardware and software engineering tools, it is possible to configure the safety control system in various ways to meet the system safety requirements. This has made it possible to minimize the cost of the entire control system to a great extent.

Moreover, safety PLCs are tested for certification of agreement to the safety standards from national research institutes or safety inspection companies such as TÜV[29]. TÜV Group is a European Notified Body, authorized to certify to EU Directives (Machinery, Elevators, Low-Voltage, Pressure equipment and EMC) in European Union, US and Canada. TÜV Industrial Services GmbH, Automation, Software and Information Technology, (ASI) has a testing laboratory which is accredited for various operating ranges by different organizations. ASI performs type-approval testing and certifications on behalf of the manufacturer. The purpose is to ensure that the manufactured product includes sufficient functional safety according to the intended Safety Integrity Levels (SIL). TÜV Group takes the requirements of national and international standards such as IEC 61508 into consideration and ensures the product's suitability for the intended application area. The services of TÜV are used worldwide by business and government contractors.

The integration of safety and automation becomes an easy task with safety PLC. The automation design engineers are taking advantage of specially designed safety PLCs:

- Save startup costs.
- Reduce downtime.
- Improve flexibility of the system design.
- Improve maintenance and operational capabilities.
- Reduce damage recovery costs in case of industrial accident.
- Improve productivity by providing safe work environment for equipment and workers.

- Eliminate hardwiring safety circuits which equates to savings in wiring, engineering and maintainability.

IEC 61508 guidelines also suggests designing the automation system by using safety certified components. This guideline assures that all possible faults, random errors and systematic failures have been reviewed to achieve the required safety integrity level. The implementation of the automation system can be well documented with all its hazard analysis, design revisions, document change control revisions, approvals, test results, drawings, specification and verification and validations. All of the above are applicable for hardware components as well as software.

Chapter 3

Problem Statement

3. Problem Statement

Most of the hazard analysis methods are applied to the process industries. However, there is an opportunity to extrapolate these concepts to automation systems. IEC 61508 safety standard is a generic standard that provides a framework applicable to all PES based applications. PLC-based automated control systems are examples of PES systems. IEC 61508 standard mainly deals with “Functional Safety” and it also provides a guideline for the hazard analysis in order to identify possible failure modes and their consequences.

There is no reported evidence in the literature on how the results of the hazard analysis methods are used for designing the safety-based automated control systems. The aim of this work is to conduct experiments and draw conclusions on how this information can be used. This work will focus on identifying potential functional safety needs in an automated manufacturing system, using methods proposed in IEC 61508 standard. This information will be used to identify and analyze unique safety performance improvements that can be implemented using safety PLC, instead of a traditional non-safety PLC.

The implementation requirements will be derived after analyzing possible failures. These failures will be derived from functional safety needs. These failures will serve the purpose of the test scenarios. Once the failures are identified, the hazard analysis method will be applied on each failure to identify possible causes and consequences. Each cause will be analyzed independently and collectively to list the design recommendations for the safety control system. The guidelines from IEC 61508 standard will be used to establish the control system

design and achieve the desired level of safety as determined by the hazard analysis. The results of the hazard analysis will be used to compare the performance of the control system before and after the implementation of the safety system. Finally, performance of the control system will be compared for its fault response, fail-safe operation, troubleshooting capabilities and ease of safety function engineering.

The design and implementation of the safety system will take into consideration various factors such as the intended application, available safety technology, safety integrity level requirements and the cost of the system. This will enable the selection of the appropriate safety control architecture for a given application. Manufacturers of safety systems promote integrated control solutions in their product offering. These manufacturers offer both general purpose and safety controls in the same controller. This approach may not be the most effective solution for some applications. This research work uses the guideline provided by IEC 61508 standard, to implement two independent but integrated systems one for process control and one for safety. The uniqueness is in the investigation of how two independent systems, from two different manufacturers can be integrated into one application.

The implementation and the integration will be demonstrated by using a Robotic cell consists of three IBM Robotic stations, a conveyor and a DVT vision system. All equipment are monitored and controlled by a non-safety PLC and five distributed I/O modules. The RSView-based Human Machine Interface (HMI) is provided to control and monitor the operation of this assembly line remotely.

The scope of this thesis will be limited to two Robotic stations and a conveyor. These systems will be analyzed for functional safety needs. Once the failure modes with respect to each functional safety need are identified, the Failure Mode Effect Analysis and Risk-Graph Analysis will be applied to determine possible causes and consequences of each failure mode. The results of the FMEA and Risk-Graph Analysis will be used to design the specifications of the safety control system. A Siemens safety PLC series will be used to achieve the desired safety specifications. It includes Safety certified PLC, I/O modules, sensors and actuators. It also includes IEC 1131-3 certified PLC programming software with safety certified programming modules.

Two different automated control system architectures will be compared for the system performance one with the non-safety control system and other with the Siemens safety control system. Various faults will be generated in both control systems to observe the system response with respect to safety. An attribute analysis will be performed to compare these responses, and the appropriate conclusions will be derived. Furthermore, this work can be used as a guideline to analyze, design, implement and validate safety-related functional needs for an industrial manufacturing application.

Chapter 4

Experimental Setup for “Before Case”

4. Experimental Setup for “Before Case”

4.1 Description of the System

The Robotic assembly line is located in the Robotic automation laboratory in Center of Integrated Manufacturing Systems. This assembly line is designed to produce the written paper product. The materials fed to this assembly line are stacks of paper and three pens of different colors. The assembly line is designed to write three letters by using selected pen color on a paper substrate. This line is composed of three Robotic stations, a conveyor line, vision system and two material feeder stations. The system is controlled by a non-safety Ethernet based PLC with the remote I/O modules. Figure 4.1 shows the schematic of the Robotic manufacturing cell and its main components.

This assembly line is divided into seven stations.

a. **Station 0:** This is the start station. When the conveyor is started, a stopper blocks all the pallets. A proximity sensor checks for the proper orientation of the pallet. If the pallet has the proper orientation, the stopper releases it after a predefined time. If the pallet is disoriented, the stopper does not release the pallet.

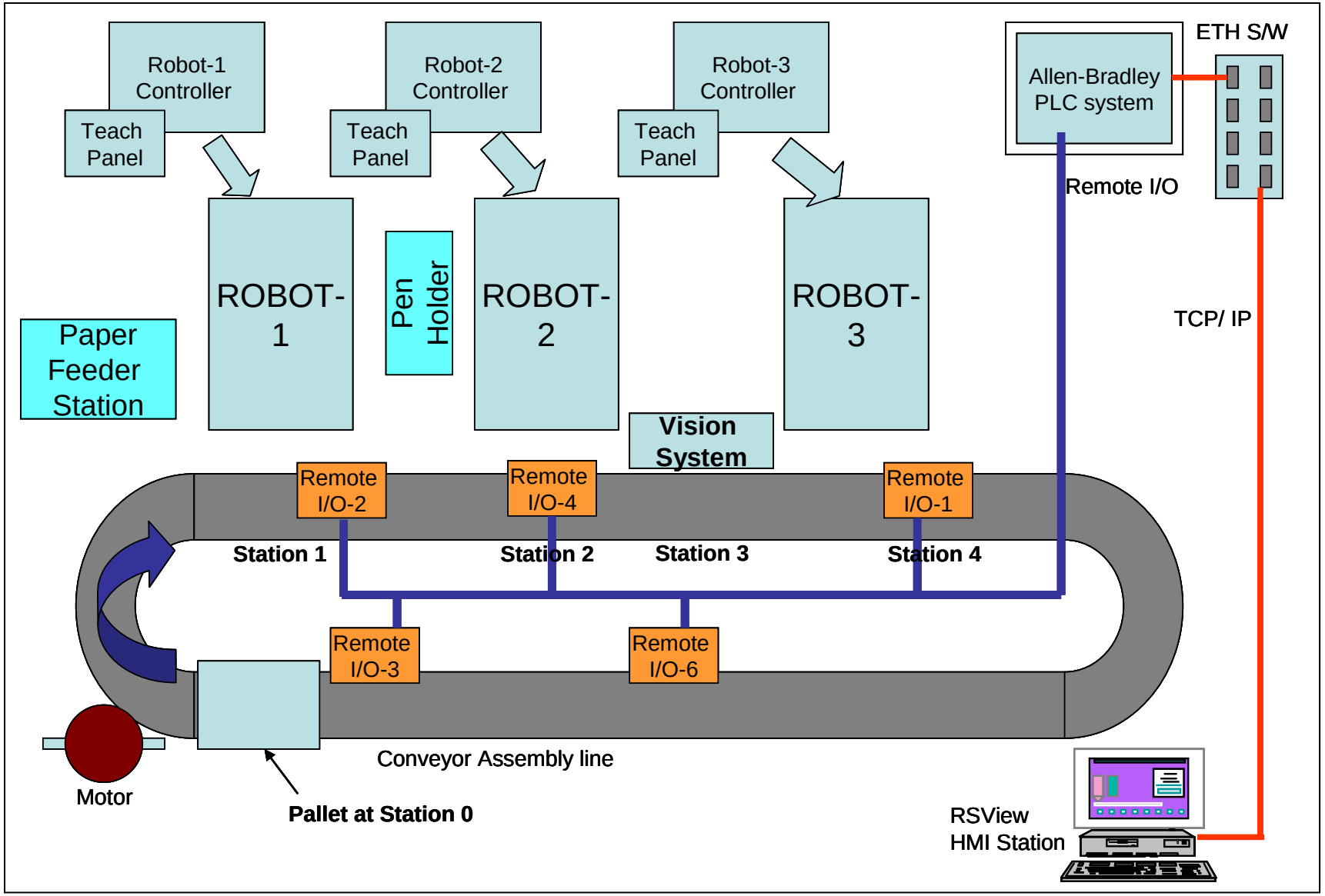


Figure 4.1: Schematic diagram of the “Before Case” Robotic Manufacturing Cell.

b. **Paper Feeder Station:** This station is built with a container which holds multiple paper stacks. A pair of photoelectric beam sensors monitors the level of paper in the container. These sensors give a paper empty signal as soon as the paper level drops below the reorder quantity. This station is installed within the Robot 1 work-envelope.

c. **Station 1:** This station is dedicated to the Robot 1 operations. As soon as the pallet arrives at this station, it is detected by a proximity sensor. This sensor activates a stopper which blocks the pallet. The stopper at station 1 remains activated when the Robot 1 is in operation or a pallet is present at station 2. After a predefined time, a clamp is actuated to hold the pallet. Simultaneously, the start signal is sent to the Robot 1 controller. This signal triggers the Robot 1 control program. The Robot 1 extends its arm to the paper feeder station and grasps a paper and places it on the pallet. At the end of this operation, Robot 1 provides a done signal which is used to release the pallet from station 1, and Robot 1 waits for the next pallet.

d. **Pen Feeder Station:** This station has a capacity to hold three pens for Robot 2 operation. Three pairs of photoelectric beam sensors detect the presence or absence of the pens. These signals are sent to the PLC for evaluation. This station is installed within the Robot 2 work-envelope.

e. **Station 2:** This station is dedicated to the Robot 2 operations. Once the pallet is released from station 1, it is detected by a proximity sensor. This activates the stopper to block the pallet. This stopper remains activated when the Robot 2 is in operation or the pallet is

present at Station 3. Once the pallet is detected, a clamp is actuated and a start signal is sent to Robot 2 controller. When the “Pen color” and “Data to Print” is entered from the HMI screen, Robot 2 picks the pre-selected pen and writes three selected characters. At the end of this operation, Robot 2 places the pen back to its original position on the pen feeder and releases the pallet.

f. Station 3: This station is dedicated to the visual inspection system. The vision system captures the image and performs an Optical Character Recognition (OCR) function. The vision system provides OCR results in a string format and sends it to the PLC. The PLC compares the OCR data with the expected outcome and generates a pass/fail signal depending on the comparison. At the end of this operation, the pallet is released from station 3.

g. Station 4: This station is dedicated to Robot 3 operations. When the pallet arrives at station 4, it is detected by a proximity sensor which activates a stopper. Simultaneously, a start signal is sent to the Robot 3. Based on the pass/fail signal generated by station 3, Robot 3 places the paper in the “good” or “bad” stack.

4.2 Components

4.2.1 Robot 1, Robot 2 and Robot 3

Introduction:

IBM 7545 Robot is an electronically-driven, microprocessor-controlled system that offers speed and repeatability for flexible automation applications. A DOS based personal computer is used as a programming device for this system. A programming language called A Manufacturing Language/ Entry (AML/ Entry) is used for writing the programs. Thus, system consists of three major components:

A. Manipulator.

B. Controller.

C. Operator Control Panel.

A. Manipulator

The manipulator is a two-jointed arm structure with four degrees of freedom. The joints of the arm, called Theta 1 (X) axis and Theta 2 (Y) axis, provide two degrees of freedom through their swivel motion. The end-of-arm rotation, called the Roll axis (r), provides a third degree of freedom. The end-of-arm also provides a fourth degree of freedom through a vertical motion (Z axis). The end-of-arm provides air connection to a pneumatic gripper.

The end-of-arm mounting head, called end-effector is removable so different types of fixtures can be attached for various operations. The end-effector on the Robot 1 is designed to hold two vaccum cups. These cups grasp and release the paper substrate. The end-effector on the Robot 2 is designed with a mechanical fixture to grasp and release the pen.

The manipulator movement within the four degrees of freedom produces an area which is called work envelope. The work envelope is set differently for each Robot depending on its Theta 1 (X) and Theta 2 (Y) axes orientations. The work envelope for Robot 1 covers the Paper feeder station and Station 1 on the conveyor. The work envelope for Robot 2 covers the Pen feeder station and Station 2 on the conveyor.

B. Controller

The controller contains most of the electronics to control the manipulator. A microprocessor coordinates the manipulator's movement and monitors its speed and positioning. Peripheral devices are synchronized with the manipulator through the use of digital input (DI) and digital output (DO) ports. The digital input ports monitor open/ close switches (DI) external to the system and the digital output ports operate relays (DO) allowing events to occur. The controller receives and stores its control program from the DOS based personal computer and then drives the manipulator by executing the program. The application program provides the instructions to the controller as to which ports to monitor, the amount of time to wait for the event and the type of condition to expect. The three main boards inside the controllers are:

1. CPU Board: The CPU board contains a microprocessor, storage, interface circuits and communication circuits.
2. Motor Control Board: It contains the roll microprocessor and circuits to control the movement as directed by the CPU. It keeps track of movements through the use of inputs from the manipulator.

3. Relay Board: The relay board contains relays, power distribution and interface circuits.

The controller has three connectors which allow interface with external devices. These connectors are:

Connector C1: Communication port

The 25-pin D-type connector on the controller with the label C1 RS232C is the connector for the communication interface. Communication between the personal computer and Robot controller is accomplished by using RS 232 interface. The asynchronous communication line protocol is used with following characteristics:

1. Full duplex transmission with a half-duplex (flip/flop) end-to-end user protocol.
2. Baud Rate = 9600.
3. Parity = None.
4. Data Bits = 8.
5. Stop Bits = 1.

The AML/ Entry software allows the selection of communication port (COM1 or COM2) for the program load/ unload. All other communication port settings are done in COM port properties of the personal computer. Once the communication is established, the commands can be sent to the Robot controller through AML/Entry command prompt.

Connector C2: DI/ DO Interface

The 54-pin connector on the controller with the label C2 is the connector for the Digital Input and Digital Output interface. Appendix 3 presents a list of the pins on the DI/ DO C2 connector and their function. These pins are used to perform regular DI/ DO operations as well as pre-defined functions.

Digital Inputs:

There are 16 general purpose digital inputs and 5 control inputs. Six out of 16 general purpose inputs are also used as command digital inputs. A "1" at digital input indicates a connection to DI ground and a "0" at digital input indicates no connection to DI ground.

The following Control Inputs are used in this application:

a. Inhibit Move to Home: This point pin W, when connected to DI ground, inhibits movement to the home position.
b. Emergency Stop: This point pin X must be connected to DI ground before manipulator power can be turned on.
c. Manipulator Power: Manipulator power is powered up by connecting the two inputs pin Y and pin Z; once the manipulator is powered up, these pins can be terminated.

The DI points from 12 to 16 are multifunction inputs. These inputs are used to send 5-bit command code and a strobe bit to the controller. The command codes can be recognized by the Robot controller only on the rising edge at the strobe input. Appendix 4 presents a list of 5-bit command codes and applicable function.

The list below presents the six command codes that are used in the current system:

- Auto Mode: This command code causes the system to enter into the Auto Mode. When the Auto Mode is activated, the Manual Mode DO point is turned off.
- Reset Error: This command code resets an error condition. When the error has been corrected and the error condition is reset, the Error DO is turned off.
- Return Home: This command code causes the manipulator to return to the home position. When this position is reached, the At home DO point is turned on.
- Select Application 1: This command code selects the application program downloaded in partition 1.
- Start Cycle: This command code starts the application cycle when the system is in Auto Mode with an application selected. Once the cycle starts, the cycle running DO is turned on.
- Command Strobe: When pin a is connected to DI ground, a strobe is sent to the controller. The command code is sent by OFF to ON transition of this DI point. The proper command code is issued before issuing the Command Strobe rising pulse.

Digital Outputs:

There are ten general purpose digital outputs, four status outputs and six command status outputs. When the DO point is “1,” it closes the connection to the load and when the DO point is “0,” it opens the connection to the load.

The following Status Outputs are used in this application:

- Cycle Running: This point pin e is on when in Auto Mode, and an application has been selected and started.
- Error: This point pin f is on during an error condition. This error is generated by the controller due to the internal fault conditions such as communication error, data error and overrun condition.
- At Home: This point pin g is on when the manipulator is at the HOME position.
- Unable to Move Home: This point pin h is on when the Return Home function is activated and the Inhibit Move to Home DI pin W is connected to DI ground.

The DO points from 11 to 16 are multifunction outputs. These points are also used for following command status signals:

- Manipulator Power On: This point indicates that the manipulator power is on.
- Online: This point indicates that the remote communications are enabled. When online, the system responds to all commands sent over the communication line as well as the commands sent over the Remote Operator Control Panel.
- Manual Mode: This point indicates that system is in manual mode. Manual mode can only be entered by pressing the Manual mode button on the operator control panel.
- Cycle Stopping: This point indicates that an application program is in the process of stopping, after the STOP Cycle command is sent. Once the cycle has stopped, the DO is turned off.
- Overtime: This point indicates an overtime condition and remains on until the Error Rest function is invoked.

- Op Panel Disabled: This point indicates that the operator control panel is disabled.

Connector C3: Operator Control Panel Interface

The 54-pin connector with the label C3 is used as an interface between the Robot controller and the Operator Control Panel.

C. Operator Control Panel

The control panel provides a way to control the manipulator by issuing commands through pressure-sensitive keys. It also provides a means of monitoring the operation of the system by observing which LEDs are lit. The operator control panel can be used to control the movement of the Robot, when the C2 connector (DI/ DO interface) is not connected to any peripheral device such as PLC. This panel can also be used to teach multiple points within the work envelope that are necessary for Robot programs.

4.2.2 PLC Architecture

The PLC based control system is designed for controlling Robot 1, Robot 2, Robot 3 and a Conveyor. The PLC based control system is implemented in distributed control architecture. The CPU PLC-5/40E has a remote I/O scanner port and an Ethernet port. Four remote I/O modules are connected to the CPU over the Remote I/O bus. This architecture provides the flexibility of connecting I/O modules near each Robot station and on the conveyor. The distributed system reduces the wiring complexity to a greater extent. Figure 4.2 shows the distributed PLC based control architecture for a “Before Case.”

Each I/O module (Rack) is dedicated to a specific functional area. For example Rack 2 is dedicated to Robot 1, Rack 4 is dedicated to Robot 2, Rack 6 is dedicated for proximity sensors on conveyor and photo beam sensors on pen feeder station and Rack 3 is dedicated to all the pneumatic controls (clamps and stoppers) on the conveyor.

The PLC-5/40E rack includes CPU, Co-processor module, Power supply and DH+/DH-485 adapter. The CPU has a Remote I/O scanner port at Ch 1 B. All the I/O racks are connected over the Remote I/O bus in multidrop network. It has a three-wire connection connected to each I/O module at Blue, Clear and Shield terminals. The baudrate for this remote I/O bus is set to 57.6 KBaud. Each I/O module is configured with a unique address. The CPU also has an Ethernet port which is configured for IP address: 129.21.92.67. The CPU acquires the I/O signal status over the Remote I/O bus and supplies this information over the 100 Mbps Ethernet network. The HMI application is designed with RS View 32 and is also connected over the same Ethernet network. The PLC- 5/40E and HMI communicate over Ethernet network.

Appendix 5 presents the PLC I/O list that shows the interconnection between the robots, Sensors, actuators and PLC I/O modules. Appendix 6 shows the wiring schematic for the connection between Robot I/Os and PLC I/Os.

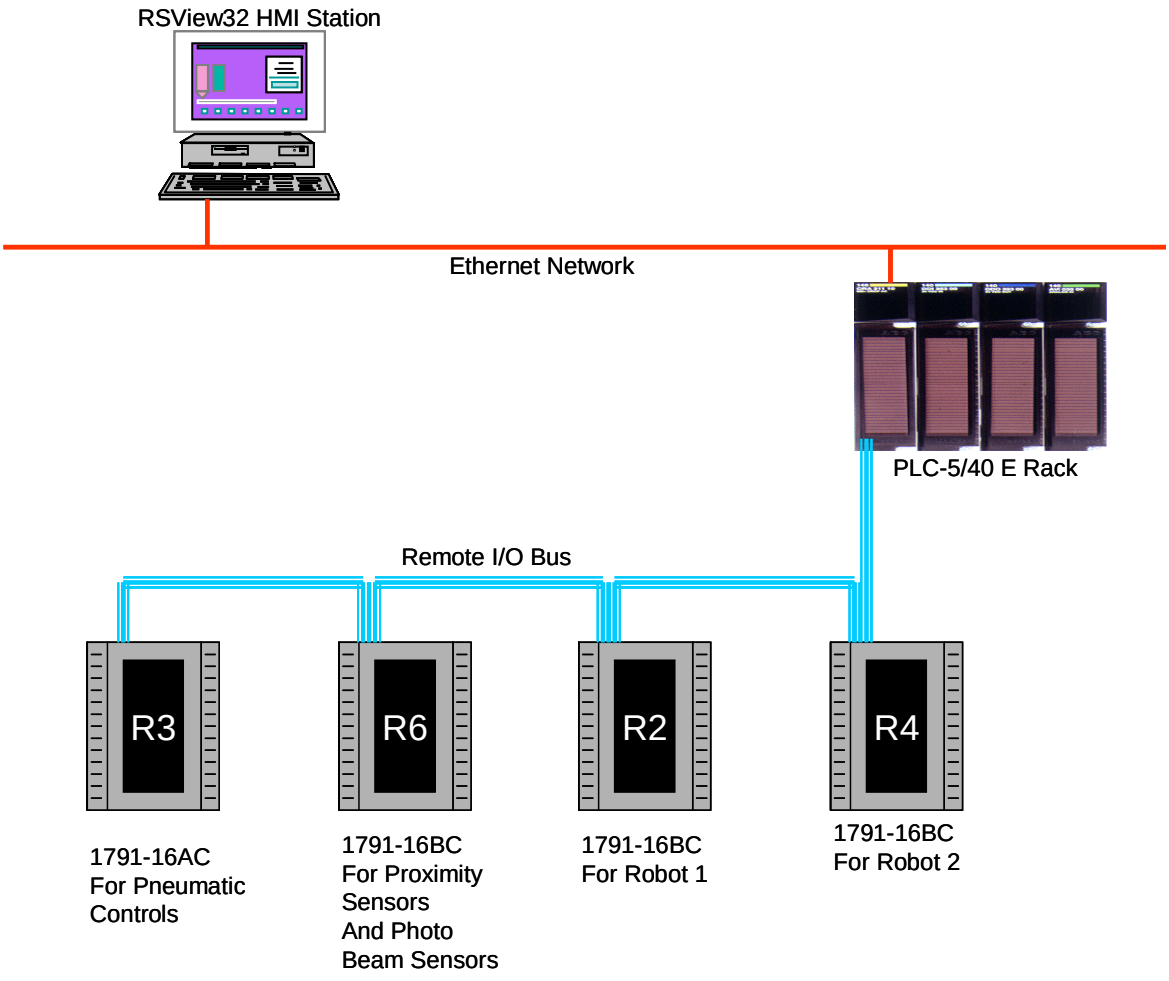


Figure 4.2: The distributed PLC based control architecture for “Before Case” system.

4.2.3 Sensors and Actuators

The system has six proximity sensors, four photobeam sensors, eight pneumatic actuators and a 120VAC Motor. Appendix 7 presents the list of the sensors and actuators and their location. Appendix 8 presents the wiring schematic of the motor control operation. The Start

and Stop pushbuttons are used to control the motor operation. There is no hardwired ESTOP function implemented in this system for Robot 1, Robot 2 and a conveyor.

4.3 Programming

4.3.1 Robot Programming

A Manufacturing Language/ Entry (AML/ Entry) version 4.1 is used for Robot programming. The command set is listed in the IBM 7545 series AML programming guide. AML/ Entry application software (.exe) is run from a DOS command prompt. This application software is used:

- To establish the communication between a personal computer and a Robot.
- To write and edit the program.
- To compile the program.
- To load/ unload the program to/ from a Robot.

Each Robot has a specific program. Robot 1 is programmed to pick up a paper substrate and place it on the pallet present at station 1. Robot 2 is programmed to pick up a selected pen and write three characters on the paper substrate.

Appendices 9 and 10 explain the logic for both Robot programs in flow-chart format and appendices 11 and 12 present the actual programs.

4.3.2 PLC Programming

The RSLogix 5 software is used for PLC programming. All the programs are developed in Ladder programming language. The entire program is divided into main program and subroutines.

The main program is developed to perform the following functions:

- Call the Robot 1 startup subroutine.
- Call the Robot 2 startup subroutine.
- Control the conveyor operation.
- Read the status of proximity sensors on the conveyor.
- Control the pneumatic stoppers and clamps.
- Call the Pen Check subroutine.
- Call the Check Data subroutine.
- Call the Send Char subroutine.
- Execute the data reset function in case of emergency stop of the Robot 2.

Appendices 13, 14, 15, 16, 17 and 18 present the associated flow charts and functional details for the main program and subroutines.

4.3.3 HMI Programming:

The RSView 32 application software is used to develop the Human Machine Interface. Table 4.1 explains the list of screens and their functions.

No.	HMI Display Screen	Functions
1	Main Screen	• Start the project • Navigate to the Station 1 screen • Navigate to the Station 2 screen • End the project
2	Station 1	• Switch Robot 1 Manipulator Power ON • Control the ESTOP for Robot 1 • Displays the current status of the Station 1 • Navigate to the Robot 1 status screen • Navigate to the Main screen
3	Station 2	• Switch Robot 2 Manipulator Power ON • Switch on the Conveyor • Control the ESTOP for Robot 2 • Navigate to the Robot 2 status screen • Displays the current status of the Station 2 • Navigate to the Data Entry screen • Navigate to the Main screen
4	Station 1 Status	• Display the current status of the Robot 1 • Navigate to the Station 1 screen
5	Station 2 Status	• Display the current status of the Robot 2 • Navigate to the Station 2 screen
6	Data Entry	• Enter character to print data • Select the pen • Display available pen color • Click “Print” to send the information to the PLC • Navigate to the Station 2 screen

Table 4.1: List of HMI display screens and associated functions.

Chapter 5

Test Cases, Risk Assessment and Selection of Safety Control System Architecture

5. Test Cases, Risk Assessment and Selection of Safety Control System Architecture

5.1 *Introduction*

The “Before Case” Robotic manufacturing cell is controlled by a non-safety PLC-based control system. This system is operated in three modes:

1. Normal Operation.
2. Teaching.
3. Maintenance.

Normal Operation:

The scope of Normal Operation of the system includes fully automated operation of the robots and the conveyor using the PLC based control system. It does not include the controlling of robots by using operator control panel and/ or operating the PLC in programming mode.

Teaching:

The scope of the Teaching mode includes operation of the robots using the Robot command prompts or by using operator control panel.

Maintenance:

The scope of the Maintenance mode includes testing of the PLC I/Os, Robot I/Os, PLC programs, Robot programs and HMI programs. It also includes testing or repairs of the hardware connection and component wiring.

The system response in case of fault event is analyzed for all operation modes.

5.2 *Selection of Test Cases*

There are many fault events possible with each mode of operation. The current control system does not have the capability to detect these fault events. These fault events may result in a minor or severe injury to personal and damage to the product or the system. Although the current system is smaller in scope compared to industrial applications, this system can be used to model real-life scenario. Each component of the system can be analyzed for different failure modes to identify all possible fault events. In a real-life scenario, a team of experts works together to analyze and find out all the possible functional safety needs of the system. The performance of each component can be analyzed for safety under different test conditions, such as various combinations of functions and operating modes.

It is not feasible to cover all possible fault events under the scope of this thesis due to the limited time and resources. For this reason, four events are selected to represent the functional safety needs of the current system. These test cases are selected such that each of

the four cases covers a fault event associated with each operating component connected in the system. In this thesis, fault conditions related to the automation system failure such as intrusion into the hazardous area, hardware failure and hardwired connection errors are considered. Mechanical system faults and software/programming errors are not considered in this study. Case 1 represents the hazardous condition associated with Robot 1 and its operation. Case 2A represents the hazardous condition associated with Robot 2 operation and the operator's access to its work envelope. Case 2B represents the hazardous condition due to the hardware failure of a component such as wire break or a relay failure, which can happen in Robot 1 or Robot 2. Case 3 represents the hazardous condition due to high voltage electrical connections and the conveyor rotation. To avoid real-life injuries, the response of the system was observed by forcing each fault condition using inanimate objects. These cases are used to demonstrate the calculation the safety requirements in terms of Safety Integrity Levels (SIL), design and implementation the safety control system and validation the safety system performance with respect to the identified safety needs. Table 5.1 shows the test cases selected for safety system implementation.

No.	Fault Condition	Mode of Operation	Response of the existing system	Expected response of the safety system
1	Operator enters the Robotic work cell near the Robot 1work envelope	Normal Operation	• Fault not detected • Does not check for error acknowledgment or reset	• Detects the presence of the operator in Robot 1 work envelope • Waits for error clear signal • Restarts the operation from Home
		Teaching	• Fault not detected • Does not check for error acknowledgment or reset	• Detects the presence of the operator in Robot 1 work envelope • Generates the alarm
		Maintenance	• Fault not detected • Does not check for error acknowledgment or reset	• Detects the presence of the operator in Robot 1 work envelope • Stops the Robot 1 • Waits for error clear signal to restart
2A	Operator intrusion in the station 2 pallet area	Normal Operation	• Fault not detected • Does not check for error acknowledgment or reset	• Detects the intrusion • Stops the Robot 2 • Waits for error clear signal • Restarts the operation from Home
		Teaching	• Fault not detected • Does not check for error acknowledgment or reset	• Detects the intrusion • Generates the alarm
		Maintenance	• Fault not detected • Does not check for error	• Detects the intrusion • Generates the alarm

			acknowledgment or reset	• Possible to disable it temporarily
2B	Robot 2 ESTOP relay failure	Normal Operation	• Robot 2 is still in operation unexpectedly • Failure is not detected • Diagnostics not available • Does not check for error acknowledgment	• Detects the failure of the signal through wire break circuit test • Voting logic can be implemented to improve the reliability of the signal • Diagnostic information is used to troubleshoot the fault
		Maintenance	• Robot 2 does not start • Diagnostics not available • Difficult to troubleshoot this fault • Does not check for error acknowledgment	• Diagnostic information is available to troubleshoot the fault
3	Operator reaches near the rotating motor and running conveyor	Normal Operation	• Fault not detected • Does not check for error acknowledgment or reset	• Detects the presence of the operator through door gate • Access control such as gate switch protects the operator • Stops the motor and conveyor
		Maintenance	• Fault not detected • Does not check for error acknowledgment or reset	• Detects the presence of the operator through door gate • Access control such as gate switch protects the operator • Stops the motor and conveyor

Table 5.1: Selected test cases for safety implementation.

5.3 *Risk Assessment*

5.3.1 Introduction

According to the IEC 61508 part 5[28], risk assessment techniques determine the tolerable risk for a specific situation that has to be achieved by necessary risk reduction. The safety-related systems are designed to reduce the probability of occurrence of the hazardous event and/ or the consequences of the hazardous event. The tolerable risk depends on many factors such as severity of injury, the number of people exposed to danger, the frequency at which a person or people are exposed to danger and the duration of the exposure. The inputs required to estimate the tolerable risk are:

- Guidelines from the appropriate safety regulatory authority.
- Discussions and agreements with the different parties involved in the application.
- Industry standards and guidelines.
- The best independent industrial, expert and scientific advice from advisory bodies.
- Legal requirements, both general and those directly relevant to the specific application.

As discussed earlier, there are two methods for risk assessment, quantitative method and qualitative method.

The quantitative method can be applied when:

- The tolerable risk is to be specified in a numerical manner.
- Hardware failure data is provided by the component manufacturers.

- Numerical targets have been specified for the safety integrity levels for the safety-related systems.

In this research, the hardware failure data is not available from the manufacturers and the numerical targets are also not specified for the system. Hence, the qualitative methods such as Risk-Graph and function-based FMEA are used to calculate the safety integrity level requirement for each test case.

5.3.2 Risk Assessment by using Qualitative Methods

A. Risk-Graph Analysis

IEC 61508- part 5 Annex D[30] describes a Risk-Graph analysis method and its application. According to this method, the risk is calculated using a formula 5.1;

$$R = f * C \qquad \text{----------(5.1)}$$

Where, R is the risk with no safety-related system is in place, f is the frequency of hazardous event with no safety-related system in place and C is the extent of damage. The frequency of hazardous event depends on three factors, which include the duration of stay of a person in the dangerous area (A), aversion of danger (G) and probability of occurrence (W). Figure 2.5 presents various levels of these factors and figure 2.6 explains how to determine SIL levels based on the levels of these factors. Based on these figures, Risk-Graph method is applied to each test case to determine the SIL requirement.

A fault condition, operator enters the Robotic work cell near the Robot 1 work envelope, is analyzed for test case 1. Table 5.2 shows the Risk-Graph analysis and SIL calculation for test case 1.

Variables	**Value**	**Level**
Extent of damage, C	C_b	Severe injury or death of a person
Duration of stay of a person in the dangerous area, A	A_b	Frequent to permanent
Aversion of danger, G	G_b	Hardly possible
Probability of occurrence, W	W_3	Relatively high
SIL level required	**SIL 3**	

Table 5.2: Risk-Graph analysis for test case 1.

Based on the above data, the SIL 3 certified safety-related system is required to avoid any hazardous event related to this fault condition.

A fault condition, operator intrusion in the station 2 pallet area, is analyzed for test case 2A. Table 5.3 shows the Risk-Graph analysis and SIL calculation for test case 2A.

Variables	**Value**	**Level**
Extent of damage, C	C_b	Severe injury or death of a person
Duration of stay of a person in the dangerous area, A	A_b	Frequent to permanent
Aversion of danger, G	G_b	Hardly possible
Probability of occurrence, W	W_3	Relatively high
SIL level required	**SIL 3**	

Table 5.3: Risk-Graph analysis for test case 2A.

Based on the above data, the SIL 3 certified safety-related system is required to avoid any hazardous event related to this fault condition.

A fault condition, Robot 2 ESTOP relay failure, is analyzed for test case 2B. Table 5.4 shows the Risk-Graph analysis and SIL calculation for test case 2B.

Variables	**Value**	**Level**
Extent of damage, C	C_b	Severe injury or death of a person
Duration of stay of a person in the dangerous area, A	A_b	Frequent to permanent
Aversion of danger, G	G_b	Hardly possible
Probability of occurrence, W	W_2	Low
SIL level required	**SIL 2**	

Table 5.4: Risk-Graph analysis for test case 2B.

Based on the above data, the SIL 2 certified safety-related system is required to avoid any hazardous event related to this fault condition.

A fault condition, operator reaches near the rotating motor and running conveyor, is analyzed for test case 3. Table 5.5 shows the Risk-Graph analysis and SIL calculation for test case 3.

Variables	**Value**	**Level**
Extent of damage, C	C_b	Severe injury or death of a person
Duration of stay of a person in the dangerous area, A	A_b	Frequent to permanent
Aversion of danger, G	G_b	Hardly possible
Probability of occurrence, W	W_3	Relatively high
SIL level required	**SIL 3**	

Table 5.5: Risk-Graph analysis for test case 3.

Based on the above data, the SIL 3 certified safety-related system is required to avoid any hazardous event related to this fault condition.

Based on the Risk-Graph Analysis, it can be seen that three of the four test cases need SIL-3 safety-related system to operate in safe condition.

B. Function-based FMEA

Another qualitative risk assessment method called Function-based FMEA is used to calculate the Probability of occurrence, Severity and possibility of detecting the failure in each test case. This method is used to verify the safety-level requirement generated by a Risk-Graph analysis. The function-based FMEA analysis includes:

A. Function/ Requirement: The intended operation designed for the system. Any deviation from the intended function results in a functional failure.
B. Potential Failure Modes: It describes the deviation from the intended function.
C. Potential Causes of Failures: These are the reasons or root causes for deviations.
D. Occurrence: Probability at which this failure occurs.
E. Local Effects: These are the immediate effects of the failures within the product or the system being analyzed and may not be recognized by the user.
F. End Effects: These are the noticeable effects on product, the system and a user.
G. Severity: It describes the seriousness of the potential end effect.
H. Detection Method/Current Controls: It describes the possibility of detecting the failure before it creates the local effects.

The Risk Priority Number (RPN) is a product of Probability of Occurrence (O), severity of the failure (S) and the detectability of the failure (D). The RPN value provides an estimated

level of intensity of the hazardous condition associated with the failure. Higher the RPN value associated with the failure, higher priority is placed to reduce the risk associated with the cause of failure. As the priority level of risk increases, the safety requirement increases to reduce the risk associated with the cause of failure. The level of safety requirements can be linked to the RPN values as follows:

- RPN > 900 requires highest level of safety.
- 900 >= RPN > 700 requires high level of safety.
- 700 >= RPN > 400 requires moderate level of safety.
- 400 >= RPN >= 100 requires low level of safety.
- RPN < 100 requires very low level of safety, no special safety is required.

The function-based FMEA analysis for each test case with “Before Case” system is shown in Table 5.9. The index ratings for probability of occurrence, severity and detection for each test case are calculated based on the article on Failure Modes and Effects Analysis by S. Kmenta and K. Ishii[31]. Table 5.6 presents the index ratings of probability of occurrence, Table 5.7 presents the index ratings of severity and Table 5.8 presents the index ratings of likelihood detection used for the FMEA analysis.

Probability of Failure	Possible Failure Rates	Occurrence Ranking
Very High: Failure is almost inevitable	>= 1 in 2	10
	1 in 3	9
High: Repeated failures	1 in 8	8
	1 in 20	7
Moderate: Occasional failures	1 in 80	6
	1 in 400	5
	1 in 2000	4
Low: Relatively few failures	1 in 15000	3
	1 in 150000	2
Remote: Failure is unlikely	1 in 1500000	1

Table 5.6: Index ratings of the probability of occurrence[31].

Effect	Severity of Effect	Severity Ranking
Hazardous without warning	When a failure mode affects safe device operation without warning	10
Hazardous with warning	When a failure mode affects safe device operation with warning	9
Very high	Device inoperable: loss of primary function	8
High	Device operable: at a highly reduced level of performance	7
Moderate	Device operable: at a reduced level of performance	6
Low	Device operable: at a slightly reduced level of performance	5
Very low	Device operable: defect noticed by most	4

	customers	
Minor	Device operable: defect noticed by average customers	3
Very minor	Device operable: defect noticed by discriminating customers	2
None	Almost no effect	1

Table 5.7: Index ratings of the severity of effect[31].

Likelihood of Detection	**Detection Ranking**
Almost impossible to detect	10
Remote detection	9
Very slight detection	8
Slight detection	7
Low detection	6
Medium detection	5
Moderate chance of detection	4
High probability of detection	3
Very high probability of detection	2
Almost uncertain to detect	1

Table 5.8: Index ratings of the likelihood of detection[31].

Sr. No.	Function/R-equirement	Potential Failure Modes	Potential Causes of Failure	Occurrence (O) (1-10)	Local Effects	End Effects on Product, User, other systems	Severity (S) (1-10)	Detection Method/ Current Controls	Detection (D) (1-10)	RPN = (O x S x D) (1-1000)
1	Robot 1 work envelope	Operator hit by Robot arm	Operator intrusion in Robot 1 work envelope	10	Hit by a Robot 1 arm	Generates Robot 1 error (moves out of work envelope, severe injury to the operator, unintentional dropping of the substrate because of the air pressure release	9	No Detection	10	900
			Operator trying to empty or change paper container	10	Hit by a Robot 1 arm	Generates Robot 1 error (moves out of work envelope, severe injury to the operator, unintentional dropping of the substrate because of the air pressure release	9	No Detection	10	900

2A	Robot 2 Work envelope	Operator hit by Robot arm	Operator intrusion in Robot 2 work envelope	10	Hit by a Robot 2 arm	Generates Robot 2 error (moves out of work envelope, severe injury to the operator, unintentional dropping of the pen because of the air pressure release	9	No Detection	10	900
		Pinch point at station 2 pallet area	Operator trying to adjust a pallet or a substrate on a pallet	10	Pinch points due to Robot's movement in Z axis	Generates Robot 2 data error, severe injury to the operator because of the forceful movement of the Robot's end effector	9	No Detection	10	900
		Cut by end effector tool at station 2 pallet area	Operator trying to adjust the pen in the end effector	10	Cut by the tool	Generates Robot 2 data error, severe injury to the operator because of the forceful movement of the Robot's end effector	9	No Detection	10	900
			Operator trying to chang or restock the pen in the pen feeder	10	Cut by the tool	Generates Robot 2 data error, severe injury to the operator because of the forceful movement of the Robot's end effector	9	No Detection	10	900

2B	Robot 2 ESTOP operation	Signal not recognized by Robot	Relay failure	5	Hit by a Robot 2 arm	System running in unsafe state, severe injury to the operator	9	LED indication on the relay	9	405
			PLC output channel failure	5	Hit by a Robot 2 arm	System running in unsafe state, severe injury to the operator	9	LED indication on the PLC output module	9	405
			Wire break	6	Hit by a Robot 2 arm	System running in unsafe state, severe injury to the operator	9	No Detection	10	540
3	Move the pallet on the conveyor	Operator hurt by the rotating parts of the motor	Open moving parts of the rotor	10	Pinch points	Severe injury to the operator, motor maynot rotate	8	No Detection	10	800
			Open gearbox	10	Multiple cuts	Severe injury to the operator	4	No Detection	10	400
			Entanglement of a cloth in moving conveyor	6	Entangle ment	Severe injury to the operator, conveyor may not move	8	No Detection	10	480
			Electrical shock due to live AC wiring	8	Severe electric shock to the operator	Severe injury to the operator	9	No Detection	10	720

Table 5.9: Function-based FMEA for “Before Case” System.

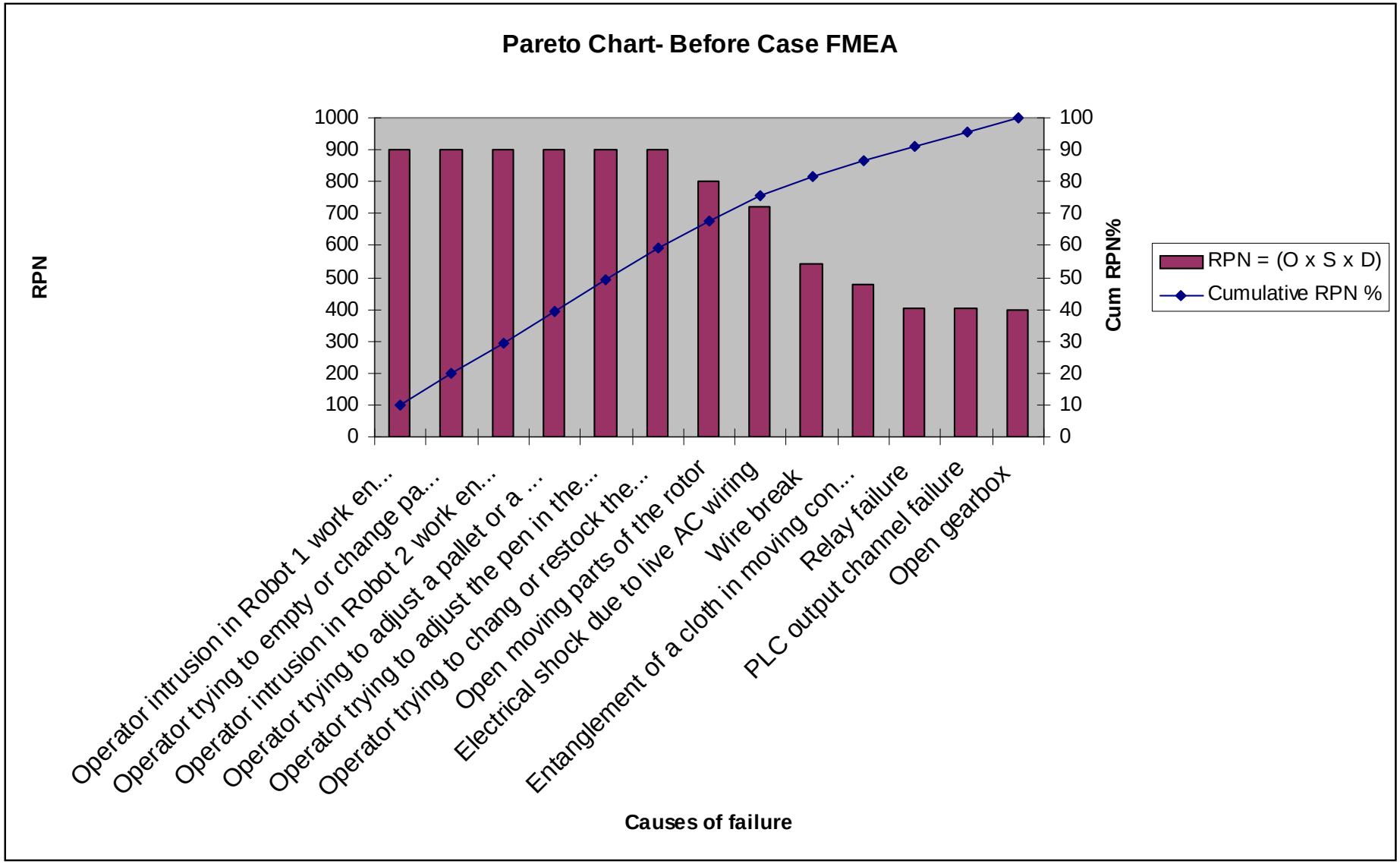


Figure 5.1: Pareto-Chart for “Before Case” FMEA Analysis.

5.3.3 Qualitative Analysis Results Evaluation

Based on the FMEA results, it can be stated that the Robot 1 and Robot 2 work envelope functions require the highest level of safety. The motor and conveyor system requires moderate-to-high level of safety, and Robot 2 ESTOP function needs moderate level of safety. When each functional failure is divided into multiple causes of failures, the safety level requirement varies depending on the probability of occurrence, severity and detection capabilities. A Pareto-Chart analysis is performed to obtain a clear picture on the safety level requirements for every potential cause of failure. The Pareto-Chart as shown in figure 5.1 shows the relationship between the Risk Priority Number and a cause of failure. Based on the Pareto-Chart, it can be stated that no safety-related controls are available in the current system to perform safety functions. A Pareto-Chart shows that 59% of total causes generate high level of hazardous condition with 900 RPN. The causes of failure included in this 59% are mainly related to Robot 1 and Robot 2 operations. Thus, to reduce the possible hazardous event, a safety-related system with high-level of safety is required for Robot 1 and Robot 2 operations. This verifies the calculated SIL 3 levels for test case 1 and test case 2A by using Risk-Graph method.

The next highest RPN is calculated for motor-related failures such as open moving parts of the motor and electrical shock due to the AC wiring. Due to highest probability of occurrence and no possible detection, high-level of safety is required to avoid the hazardous event in these cases. This verifies the calculated SIL 3 level for test case 3 by using Risk-Graph analysis. The RPN varies from 400 to 600 for remaining causes of failures. These causes require moderate level of safety. This verifies the calculated SIL 2 level for test case 2B.

Some of these causes are related to Robot 1, Robot 2 and Motor operation. Thus, safety requirements for these causes need to be raised to SIL 3 so that the SIL 3 safety requirement of Robot 1, Robot 2 and Motor can be satisfied.

5.4 *Selection of Safety Control System Architecture*

Since the safety control systems are becoming more prominent requirements in process/machine environment, many ready-to-use and customizable solutions are available. This allows the user to design precise safety control implementations, suitable for a specific application. The design of the safety control system architecture varies from simple to high level of complexity depending on the scope of the application and the SIL requirement. It is very difficult for an automation engineer to study all associated factors and select the most appropriate control system architecture which meets the requirements. Therefore, the design of the safety control system varies based on various criteria such as what kind of architectures are available, what application suits those architectures, what costs are involved and what level of safety can be achieved as per the applicable standards. A white paper published by Rockwell Automation[32] describes various factors that are considered for selecting the suitable safety control architecture.

5.4.1 Types of System Architectures

There are four types of system architectures available for designing a safety control system. These architectures include:

1. Component-based system: This is the most basic and a cost-effective way to perform low-risk safety function. For example, an E-STOP push button can be provided to cut the power to the circuit when pressed.

2. Dedicated safety relay system: A relay provides single safety function dedicated to a specific operation. It is mainly used to switch off the power to the output device when the safety input senses the fault condition. It provides basic diagnostic features such as LED indicator on the relay with input and output status. These relays are non-programmable and don't support high-level network communication interface. Although it is a simple and cost-effective solution, it can not be used for control applications which involve multiple safety functions and high level of diagnostic requirements.

3. Modular relay: This system is an expandable version of the dedicated relay. It can perform single safety function. A safety input removes the power to all output devices connected to this relay. It can handle up to 20 dual-channel inputs and 10 dual-channel outputs. It provides module-based diagnostics with network communication interface. It can not be used for multiple safety functions and for systems where safety is controlled in zones.

4. Safety PLC: The safety PLC can be used when there is a need for multiple safety functions and large number of input and output devices up to several hundreds I/Os. This is also the only solution if analog signals are involved in the system. This can be a cost-effective solution for centralized and/ or distributed architectures. The inherent

programmability and detailed diagnostics makes it an ideal choice for complex, high-risk applications.

Theoretically, a non-safety PLC with external relay circuits can be used to perform critical functions. However, such control circuits can not be certified for SIL levels and thus cannot be accepted by any safety monitoring organization.

5.4.2 Selecting a suitable system for a specific application

A thorough evaluation is necessary to understand the size and the complexity of the system before designing the safety architecture. Some safety systems may or may not address the safety requirements determined by the application. Hence, it is the automation engineer's responsibility to perform checks on the functionality requirements. For example, an ESTOP function to stop the motor can be performed by using a safety relay. When the number of I/Os increases, an additional modular relay can be added easily without oversizing the system. However, for performing functions such as timed out, muting controls and zone controls, expandable relays can not be used. In such applications, dedicated relay or safety PLC can be used. Sequential shutdown application, which requires stopping of the machine in several steps, can be performed by using safety relay or safety PLC. However, if the system needs to be shut down in multiple steps or the system control needs to be transferred from one control system to another, safety relay does not serve the purpose. In such applications, safety PLC will be the most feasible solution. The partial shutdown and zone control are few more examples where safety PLC is the only solution which can be implemented. Safety PLC is an ideal choice for process industries where hundreds of analog

signals are involved. This requirement can not be addressed by any other safety controls architecture but the safety PLC. If the system needs distributed controls and monitoring, safety PLC offers safety network interfaces to exchange data over a high-speed communication media. The signal status of the input and output devices with their diagnostic information can be made available to the controller from different locations of the plant. Dedicated safety relays provide diagnostic information in terms of the LED indicator on the module. The diagnostics information such as wire break and discrepancy error, can be generated only in safety PLC.

5.4.3 Cost factor

The cost factor is an important aspect which drives the decision of the safety control system. The goal is to get the highest level of safety achievable within the project cost. In some situations, a low investment on safety systems may cause irrecoverable loss of production, human or property damage. The scalability, the number of I/O signals, signal/equipment locations, functional requirements and affordable downtime are few criteria influence the cost of the system.

5.4.4 SIL requirement standards

A demand on the safety-related system is different for each application. Thus, two different modes of operation are defined for safety-related system in IEC 61508 standard. These are called low-demand mode and high-demand mode.

A low-demand mode is applicable to the process industry. In this mode the failure rates are defined in years. A process safety is applicable to the continuous processes such as power generation, boiler control, burner management systems and chemical plants where continuous operations are required. However, these processes are designed and controlled to stay within a normal operating envelope, and hence the demand on a safety system is low. The demand on the safety system only occurs when a specific process parameter goes outside the normal operating range. When that demand occurs, the safety system is designed to move the process to a safe state. When there is a fault in the system, it is designed to isolate the fault and keep the process running usually using redundancy. The safety system is designed also to move the process to a safe state because it can no longer monitor the specific hazard scenario reliably. The primary objective in this mode is to keep the system running continuously within a tolerable risk. The SIL levels and associated PFD ranges are shown in figure 5.2.

SIL	PFD	Max. accepted failure of SIS
SIL 1	$\geq 10^{-2}$ to $< 10^{-1}$	One hazardous failure in 10 years
SIL 2	$\geq 10^{-3}$ to $< 10^{-2}$	One hazardous failure in 100 years
SIL 3	$\geq 10^{-4}$ to $< 10^{-3}$	One hazardous failure in 1000 years
SIL 4	$\geq 10^{-5}$ to $< 10^{-4}$	One hazardous failure in 10000 years

Figure 5.2: SIL ratings for low-demand operational mode[21].

A high-demand mode is applicable to the production industry. In this mode the failure rates are defined in hours. A machine safety is applicable to the manufacturing system where CNC machines and robots are involved. These systems are protected by safeguards such as physical guarding, light curtains and pressure mats feeding into electrical interlocks. These

systems also involve very frequent human intervention. These types of applications have the potential for a frequent demand on the safety system. Therefore, high-demand or continuous-demand mode is used for such applications. Safeguards such as physical barriers are used to ensure that the operator keeps a safe distance. However, the safeguards do not necessarily protect against a fault in the machine. The primary objective is to protect the operator, the machine and the product. If an operator approaches a protected machine, the safeguards either physically prevent proximity or shutdown the machine. The SIL levels and associated PFD ranges are as shown in figure 5.3.

SIL	PFH (per hour)	Max. accepted failure of SIS
SIL 1	$\geq 10^{-6}$ to $< 10^{-5}$	One hazardous failure in 100000 hours
SIL 2	$\geq 10^{-7}$ to $< 10^{-6}$	One hazardous failure in 1000000 hours
SIL 3	$\geq 10^{-8}$ to $< 10^{-7}$	One hazardous failure in 10000000 hours
SIL 4	$\geq 10^{-9}$ to $< 10^{-8}$	One hazardous failure in 100000000 hours

Figure 5.3: SIL ratings for high-demand operational mode[21].

5.4.5 Architecture selection for “After Case”

The safety control system selection can be based on all of the above parameters. It can be single or a combination of multiple solutions. It is not necessary to use single architecture for the entire application. Multiple systems based on different architectures can be used and integrated together to meet the safety requirements of the application. The technological advancement in open networking protocols has made it easier to integrate various third-party

safety and non-safety control architectures together. This approach is used mainly to control the cost of the project.

IEC 61508 standard also recommends separating safety system from process control system. The reason to separate these systems is to keep the operation of safety function isolated from non-safety functions. The purpose is to ensure that a failure of any non-safety function will not jeopardize the operation of safety function of the system. The automation engineer can make use of this clause for building two different systems, one for safety (high cost) and another for normal process operation (low cost). This ensures that the control system is designed for required SIL within the given project cost.

The manufacturing system used for this work consists of two robots and a conveyor controlled by a PLC-based control system. Hence, it is considered to be an example of machine safety so the high-demand mode or continuous-demand mode is placed on the safety-related system.

Based on above discussion, the following factors are considered in selecting the safety PLC based control architecture:

- The overall calculated SIL (SIL-3).
- The safety-related system needs to perform multiple safety-related functions simultaneously.
- Distributed control architecture required to provide connectivity with the safety sensors that are physically distributed across the Robotic cell.

- Requirement for independent control systems to control the cost of the project.

A non-safety system is used to control the robots and conveyor and to perform normal operations. The safety system is used to perform safety-related functions only. In this case, the safety system has a priority in operation. Non-safety functions cannot be performed if the safety functional requirements are not satisfied. The primary objective of this safety system is to shutdown the faulty system whenever faults are detected.

Chapter 6

Description of the “After Case” System

6. Description of the “After Case” System

6.1 *Introduction*

A Siemens safety PLC system was used to implement the SIL-3 safety-related system in the robotic manufacturing cell. Based on conclusions derived in Chapter 5, a safety PLC with distributed control architecture was designed to cover all four test cases.

This safety system was implemented separately with the existing non-safety PLC control system. This safety system is designed to control the Robot 1 ESTOP (digital input), Robot 2 ESTOP (digital input) and power to the motor. The manipulator power for Robot 1 and Robot 2 can be turned on using RSView 32 HMI only if the fault conditions are not present and/or ESTOP 12 or ESTOP 3 or ESTOP ALL pushbuttons are not activated. Similarly, the conveyor can be turned on only if the fault conditions related to test case 3 are not present and/or ESTOP 12 or ESTOP 3 or ESTOP ALL push buttons are not activated. Appendices 19, 20, 21, 22, 23 and 24 show the display screens developed using RSView 32 HMI for the “After case” system.

For designing SIL-3 safety-related system, all components used from sensor to actuator have to be certified for SIL-3. Figure 6.1 shows various components required to implement SIL-3 control circuit and necessary SIL levels to achieve SIL-3 for the entire circuit.

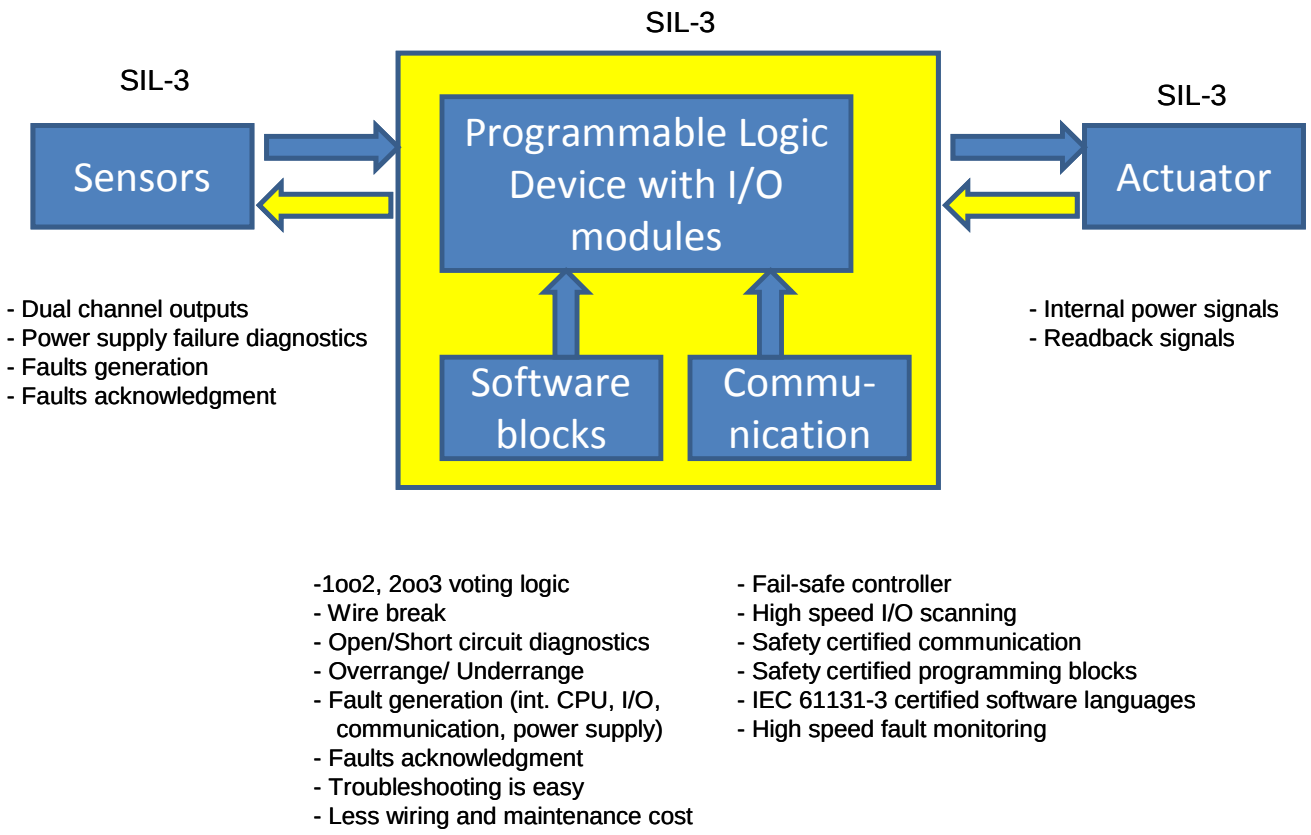


Figure 6.1: SIL-3 safety-related system components and safety integrity levels.

The implemented safety system is divided into three panels. Panel 1 consists of the safety CPU rack with safety I/O modules, Panel 2 consists of distributed safety rack with safety and general purpose I/O modules and Panel 3 is used as the operator panel to send START, STOP and Acknowledge (ACK) commands. The controls mounted on an operator panel provide functionality to start and stop all three operational components (Robot 1, Robot 2 and the motor). In the beginning, it is required to press ACK button before pressing the START button. The ACK button is provided to acknowledge the fault condition. Until the fault is acknowledged after clearing the fault condition, the stopped system can not be turned on. The non-safety PLC program is modified to remove the “software ESTOP” function.

This program now checks for the safety signals provided by the safety PLC before starting the normal operations. Three hardwired ESTOP functions are provided in this system. Appendix 25 shows the list of Siemens Safety I/O modules and Siemens safety sensors used to implement safety-related system for all four cases. Appendix 26 presents the I/O list for safety PLC and associated sensors connections. Appendix 27 shows the modified I/O list after implementation of the “After Case” system.

As shown in figure 6.1, SIL-3 safety circuits are designed by using safety-certified components. The components used are configured for safety as follows:

1. Safety Sensors:
    - All the sensors used are certified for SIL-3 configuration.
    - Dual signals are generated by these sensors.
    - These sensors are powered by using external power supply.

2. Safety Input Modules:
    - All safety input modules are certified for SIL-3 configuration.
    - Unique module address is set for each safety input module.
    - 1oo2 voting logic is used for all the safety signals.
    - Discrepancy time error is monitored for all 1oo2 connections.
    - Behavior at discrepancy time is defined.
    - Channel level fault diagnostics are enabled.
    - Behavior after channel fault is defined.

3. Safety PLC:
    - Safety CPU is certified for SIL-3 configuration.
    - Safety operating mode is activated.
    - Safety-certified software programming blocks are used.
    - OB 35, the organization block for safety, is added.
    - Safety program call structure is used via FC1.
    - Fault reintegration logic is developed in FC2.

4. Safety Output:
    - All safety output modules are certified for SIL-3 configuration.
    - Unique module address is set for each safety input module.
    - Feedback circuit is implemented through safety contactors.
    - Channel read-back time is defined.
    - Channel level wire-break fault diagnostics are enabled.
    - Behavior after channel fault is defined.

A safety-related function is designed and implemented separately for each test case. Figure 6.2 shows the implemented safety system for the robotic manufacturing cell.

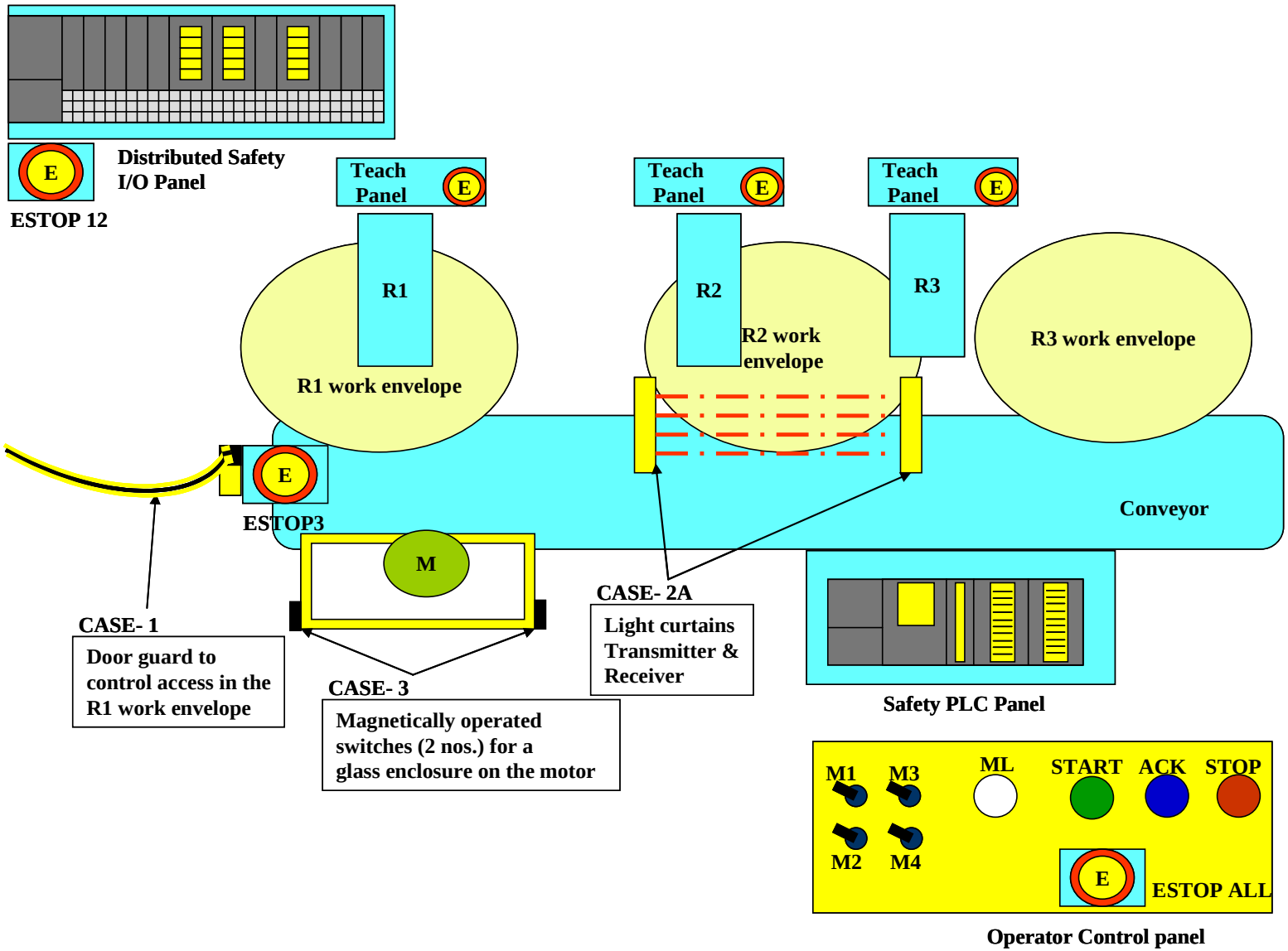


Figure 6.2: Implemented safety system architecture for the “After Case” Robotic manufacturing cell.

The safety control system implementation and its operation for each test case are discussed in the following sections:

6.2 *Description of the Safety-Related system for Test Case 1*

In test case 1, the fault condition "Operator enters the robotic work cell near the Robot 1 work envelope" is detected when the door gate switch is unplugged. Based on this fault detection signal, Robot 1 operation is controlled. A dual channel door guard safety sensor is used to detect the operator presence in the Robot 1 work envelope. Whenever this fault condition occurs, Robot 1 is turned off. The Robot 1 operation is turned off by switching the Robot 1 ESTOP digital input to 0. This digital input is driven by NO contacts of two safety contactors (K11 and K12) connected in series. The NC contacts of these contactors (K11_NC & K12_NC) are monitored in the PLC system with a logical AND gate operation and used as a feedback signal in the software program. The absence of door gate fault signal supplies the power to these two safety contactors that are connected in series. The presence of door gate fault signal disconnects the power to these safety contactors. The power to the contactors is controlled under following conditions:

- Door gate fault signal status.
- Safety contactor feedback status signal.
- Emergency stop signal activated via ESTOP 12 or ESTOP 3 or ESTOP ALL push buttons.

When the fault conditions related to test case 1 are not present, R1_healthy signal is generated by the safety system. This signal is sent to the non-safety PLC system for HMI

display. Once this signal is detected as high, only then can Robot 1 manipulator power, be turned on. Appendix 28 presents the FB1 programming block which shows the SIL-certified programming blocks used to implement test case 1 safety functions.

6.3 *Description of the Safety-Related system for Test Case 2A*

In test case 2A, the fault condition "Operator intrusion in the station 2 pallet area" is detected when the operator breaks the light barrier. Based on this fault detection signal, the Robot 2 operation is controlled. A dual channel light curtain safety sensor is used to detect the operator presence in the Station 2 pallet area. Whenever this fault condition occurs, the Robot 2 is turned off. The Robot 2 operation is turned off by switching the Robot 2 ESTOP digital input to 0. This digital input is driven by NO contacts of two safety contactors (K21 and K22) connected in series. The NC contacts of these contactors (K21_NC & K22_NC) are monitored in the PLC system with a logical AND gate operation and used as a feedback signal in the software program. The absence of the light curtain fault signal supplies the power to these two safety contactors that are connected in series. The presence of the door gate fault signal disconnects the power to these safety contactors. The power to the contactors is controlled under following conditions:

- Light curtain fault signal status.
- Safety contactor feedback status signal.
- Emergency stop signal activated via ESTOP 12 or ESTOP 3 or ESTOP ALL push buttons.

This case is implemented with MUTING operation mode. The light curtain can be muted whenever required by using four toggle switches. The toggle switches (M1, M2, M3 & M4)

are mounted on the new operator panel. In case the operator needs to load or unload pens in the pen feeder station, muting mode is activated so that even if the operator breaks the light barrier, the Robot 2 is still operational. The MUTING operation mode can be dangerous when the Robot 2 is in operation. Therefore, the muting lamp mounted on the new operator panel is turned on and remains on until the muting mode is activated. When this lamp is turned on, the pen feeder is getting restocked and hence, no data printing command is supplied to the Robot 2 from HMI. When the fault conditions related to test case 2A and 2B are not present, R2_healthy signal is generated by the safety system. This signal is sent to the non-safety PLC system for HMI display. Once this signal is detected as high, only then can Robot 2 manipulator power, be turned on. Appendix 28 presents the FB2 programming block which shows the SIL-certified programming blocks used to implement test case 2A safety functions.

6.4 *Description of Safety-Related system for Test Case 2B*

In test case 2B, the fault condition “Robot 2 ESTOP relay failure” is detected because of the wire-break diagnostics available in the safety output module. The safety-related system to perform this safety function is implemented with the test case 2A for Robot 2 and with the test case 1 for Robot 1. The NC contacts, K11_NC & K12_NC in case of Robot 1 and NC contacts, K21_NC & K22_NC in case of Robot 2 are connected to the digital input modules of safety PLC. These signals are read and monitored through SIL-certified programming block, F_Feedback. The Robot ESTOP relay circuit is replaced with the two safety contactors. The Robot ESTOP digital input is now read through NO contact instead of NC contacts. With this type of connection, the ESTOP relay failure condition is avoided. A wire-

break diagnosis is provided through the safety digital output module. A fault signal is generated in case of wire-break. This fault signal generates a fault in safety CPU and stops the Robot operation. The wire-break diagnosis information is also available in the CPU. This information is displayed in S7 software project and is used for troubleshooting such faults easily. Appendix 28 presents the OB1 organization block which shows how the feedback signal is generated for Robot 1 and Robot 2. Appendices 28 and 29 show how the SIL-certified feedback programming block is used with the feedback signal generated by OB1 for Robot 1 and Robot 2.

6.5 *Description of Safety-Related system for Test Case 3*

In test case 3, the fault condition "Operator reaches near the rotating motor and running conveyor is detected" when the operator opens the glass door enclosing the motor. Based on this fault-detection signal, the motor operation is controlled. A motor is enclosed by a door. A pair of safety magnetic door switches is mounted on the opening side of the door. The door has to be opened to access the motor and its gearbox. Whenever the door is opened, a fault signal is generated by these magnetic door switches. Whenever this fault condition occurs, the Motor is turned off. The motor is driven by NO contacts of two safety contactors (K31 and K32) connected in series. The NC contacts of the same contactors (K31_NC & K32_NC) are monitored in the PLC system with a logical AND gate operation and used as a feedback signal in the software program. The closing of this door supplies the power to the motor. The opening of this door gate generates a fault signal and disconnects the power to the motor. The power to the contactors is controlled under following conditions:

- Magnetic door switches (opening or closing of the door).

- Safety contactor feedback status signal.
- Emergency stop signal activated via ESTOP 12 or ESTOP 3 or ESTOP ALL push buttons.

When the fault conditions related to test case 3 are not present, Motor_healthy signal is generated by the safety system. This signal is sent to the non-safety PLC system for HMI display. Once this signal is detected as high, only then can motor be turned on. Appendix 28 presents the FB3 programming block which shows the SIL-certified programming blocks used to implement test case 3 safety functions, and OB1 shows how the feedback signal is generated for the motor control.

Chapter 7

Safety Validation, Conclusion and Recommendations for the Future Work

7. Safety Validation, Conclusion and Recommendation for the Future Work

7.1 *Introduction*

Once the safety-related system was implemented in the Robotic manufacturing cell, it was necessary to evaluate its performance. This process is called “Safety Validation.” The four test cases identified in chapter 5 were analyzed by using Risk-Graph and FMEA analysis and compared with results presented in chapter 5.

7.2 *Comparative Analysis between “Before Case” and “After Case” Systems*

Table 7.1, 7.2, 7.3 and 7.4 show a comparative analysis of the system response for each test case. The response of the “After Case” system is compared with the response of the “Before Case” system. The response of the “Before Case” system shows the observed behavior of the robotic manufacturing cell with no safety-related system in place. The response of the “After Case” system shows the observed behavior of the robotic manufacturing cell with SIL 3 certified safety-related system. The “After Case” system responses were tested with an operator. From this comparison, it can be seen that the detection of fault signal plays an important role in minimizing the probability of occurrence and thus the possible hazardous event.

No.	Fault Condition	Mode of Operation	Response of the "Before Case" system	Expected response of the safety system	Response of the "After Case" system
1	Operator enters the Robotic work cell near the Robot 1work envelope	Normal Operation	• Fault not detected • Does not check for error acknowledgment or reset	• Detects the presence of the operator in Robot 1 work envelope • Stop the Robot 1 • Waits for error clear signal • Restarts the operation from Home	• Does not allow operator to enter directly into the Robot 1 work envelope • Operator has to open the door switch to enter the area • Opening of the door switch stops the Robot 1 operation • Acknowledgement is mandatory to clear the error and restart the Robot 1
		Teaching	• Fault not detected • Does not check for error acknowledgment or reset	• Detects the presence of the operator in Robot 1 work envelope • Generates an alarm	Same as above
		Maintenance	• Fault not detected • Does not check for error acknowledgment or reset	• Detects the presence of the operator in Robot 1 work envelope • Stops the Robot 1 • Waits for error clear signal to restart	Same as above

Table 7.1: Comparative Analysis of the "Before Case" and "After Case" System Responses in case of fault condition for test case 1.

No.	Fault Condition	Mode of Operation	Response of the "Before Case" system	Expected response of the safety system	Response of the "After Case" system
2A	Operator intrusion in the station 2 pallet area	Normal Operation	• Fault not detected • Does not check for error acknowledgment or reset	• Detects the intrusion, stops the Robot 2 • Waits for error clear signal • Restarts the operation from Home	• Does not allow operator to enter directly into the Robot 2 work envelope • If operator breaks the light barrier, Robot 2 is stopped • In case of intrusion, acknowledgement is mandatory to clear the error and restart the Robot 2 • Intrusion is only allowed during the controlled muting mode
		Teaching	• Fault not detected • Does not check for error acknowledgment or reset	• Detects the intrusion • Generates the alarm	Same as above
		Maintenance	• Fault not detected • Does not check for error acknowledgment or reset	• Detects the intrusion • Generates the alarm • Possible to disable it temporarily	Same as above

Table 7.2: Comparative Analysis of the "Before Case" and "After Case" System Responses in case of fault condition for test case 2A.

No.	Fault Condition	Mode of Operation	Response of the "Before Case" system	Expected response of the safety system	Response of the "After Case" system
2B	Robot 2 ESTOP relay failure	Normal Operation	• Robot 2 is still in operation unexpectedly • Failure is not detected • Diagnostics not available • Does not check for error acknowledgment or reset	• Detects the failure of the signal through wire-break circuit test • Voting logic can be implemented to improve the reliability of the signal • Diagnostic information is used to troubleshoot the fault	• ESTOP Relay is removed to avoid the relay failure • Wire break is detected by safety PLC and Robot 2 is stopped • Fault is generated in safety PLC and this diagnostic information is used for troubleshooting • Two safety contactors are used to improve the reliability of the output signal and to read the output signal as a feedback • Acknowledgement is mandatory to clear the error and restart the Robot 2
		Maintenance	• Robot 2 does not start • Diagnostics not available, difficult to troubleshoot this fault • Does not check for error acknowledgment or reset	• Diagnostic information is available to troubleshoot the fault	Same as above

Table 7.3: Comparative Analysis of the "Before Case" and "After Case" System Responses in case of fault condition for test case 2B.

No.	Fault Condition	Mode of Operation	Response of the "Before Case" system	Expected response of the safety system	Response of the "After Case" system
3	Operator reaches near the rotating motor and running conveyor	Normal Operation	• Fault not detected • Does not check for error acknowledgment or reset	• Detects the presence of the operator through door gate • Access control such as gate switch protects the operator • Stops the motor and conveyor	• Does not allow direct access to the motor • Operator has to open the door gate to access the motor connections and its components • Opening of the door gate stops the motor operation • Acknowledgement is mandatory to clear the error and restart the motor
		Maintenance	• Fault not detected • Does not check for error acknowledgment or reset	• Detects the presence of the operator through door gate • Access control such as gate switch protects the operator • Stops the motor and conveyor	Same as above

Table 7.4: Comparative Analysis of the "Before Case" and "After Case" System Responses in case of fault condition for test case 3.

7.3 *Qualitative Risk Assessment for "After Case"*

Once the responses are compared, risk assessment methods such as Risk-Graph Analysis and FMEA are applied to calculate the level of safety required for the "After Case" system. These risk assessments are required to ensure that all the necessary safety-related systems are implemented and no additional safety is required to operate the system in safe mode.

7.3.1 Risk-Graph Analysis

The Risk-Graph analysis is applied to the "After Case" system. Due to the 100% detection of the fault provided in the "After Case" system, the extent of damage (C) is reduced to the level that is less than the lowest level of extent of damage (C_a: Light injury of a person, small environmental damage). According to figure 2.6, no special safety is required when the C is less than C_a. This means that the safety requirement for the "After Case" is less than SIL-1, the lowest level of safety requirement that can be calculated by the Risk-Graph analysis. Therefore, it can be concluded that no additional safety is required for the robotic manufacturing cell to operate in a safe mode and the highest level of required safety is achieved in "After Case" system. To verify this conclusion, the FMEA is used.

7.3.2 Function-based FMEA

Table 7.5 shows the FMEA analysis performed for the "After Case" system. It can be observed that, due to the highest level of detection and decreased probability of occurrence, the RPN value with respect to each cause of failure is drastically reduced compared to the "Before Case" FMEA analysis. The Pareto-Chart for the "After Case" is shown in figure 7.1.

This chart shows that the RPN for all causes of failure is less than 100. Compared to the "Before Case," this number is much lower. Based on this, it is verified that the highest level of required safety is achieved in the "After Case" system, and no additional safety implementation is required to operate the robotic manufacturing cell in safe mode.

Sr. No.	Function/R-equirement	Potential Failure Modes	Potential Causes of Failure	Occurrence (O) (1-10)	Local Effects	End Effects on Product, User, other systems	Severity (S) (1-10)	Detection Method/ Current Controls	Detection (D) (1-10)	RPN = (O x S x D) (1-1000)
1	Robot 1 work envelope	Operator hit by Robot arm	Operator intrusion in Robot 1 work envelope	3	Hit by a Robot 1 arm	Generates Robot 1 error (moves out of work envelope, severe injury to the operator, unintentional dropping of the substrate because of the air pressure release	9	No Detection	1	27
			Operator trying to empty or change paper container	3	Hit by a Robot 1 arm	Generates Robot 1 error (moves out of work envelope, severe injury to the operator, unintentional dropping of the substrate because of the air pressure release	9	No Detection	1	27

2A	Robot 2 Work envelope	Operator hit by Robot arm	Operator intrusion in Robot 2 work envelope	3	Hit by a Robot 2 arm	Generates Robot 2 error (moves out of work envelope, severe injury to the operator, unintentional dropping of the pen because of the air pressure release	9	No Detection	1	27
		Pinch point at station 2 pallet area	Operator trying to adjust a pallet or a substrate on a pallet	3	Pinch points due to Robot's movement in Z axis	Generates Robot 2 data error, severe injury to the operator because of the forceful movement of the Robot's end effector	9	No Detection	1	27
		Cut by end effector tool at station 2 pallet area	Operator trying to adjust the pen in the end effector	3	Cut by the tool	Generates Robot 2 data error, severe injury to the operator because of the forceful movement of the Robot's end effector	9	No Detection	1	27
			Operator trying to chang or restock the pen in the pen feeder	3	Cut by the tool	Generates Robot 2 data error, severe injury to the operator because of the forceful movement of the Robot's end effector	9	No Detection	1	27

2B	Robot 2 ESTOP operation	Signal not recognized by Robot	Relay failure	1	Hit by a Robot 2 arm	System running in unsafe state, severe injury to the operator	9	LED indication on the relay	1	9
			PLC output channel failure	1	Hit by a Robot 2 arm	System running in unsafe state, severe injury to the operator	9	LED indication on the PLC output module	1	9
			Wire break	6	Hit by a Robot 2 arm	System running in unsafe state, severe injury to the operator	9	No Detection	1	54
3	Move the pallet on the conveyor	Operator hurt by the rotating parts of the motor	Open moving parts of the rotor	1	Pinch points	Severe injury to the operator, motor may not rotate	8	No Detection	1	8
			Open gearbox	1	Multiple cuts	Severe injury to the operator	4	No Detection	1	4
			Entanglement of a cloth in moving conveyor	1	Entanglement	Severe injury to the operator, conveyor may not move	8	No Detection	1	8
			Electrical shock due to live AC wiring	1	Severe electric shock to the operator	Severe injury to the operator	9	No Detection	1	9

Table 7.5: The FMEA analysis for the “After Case” system.

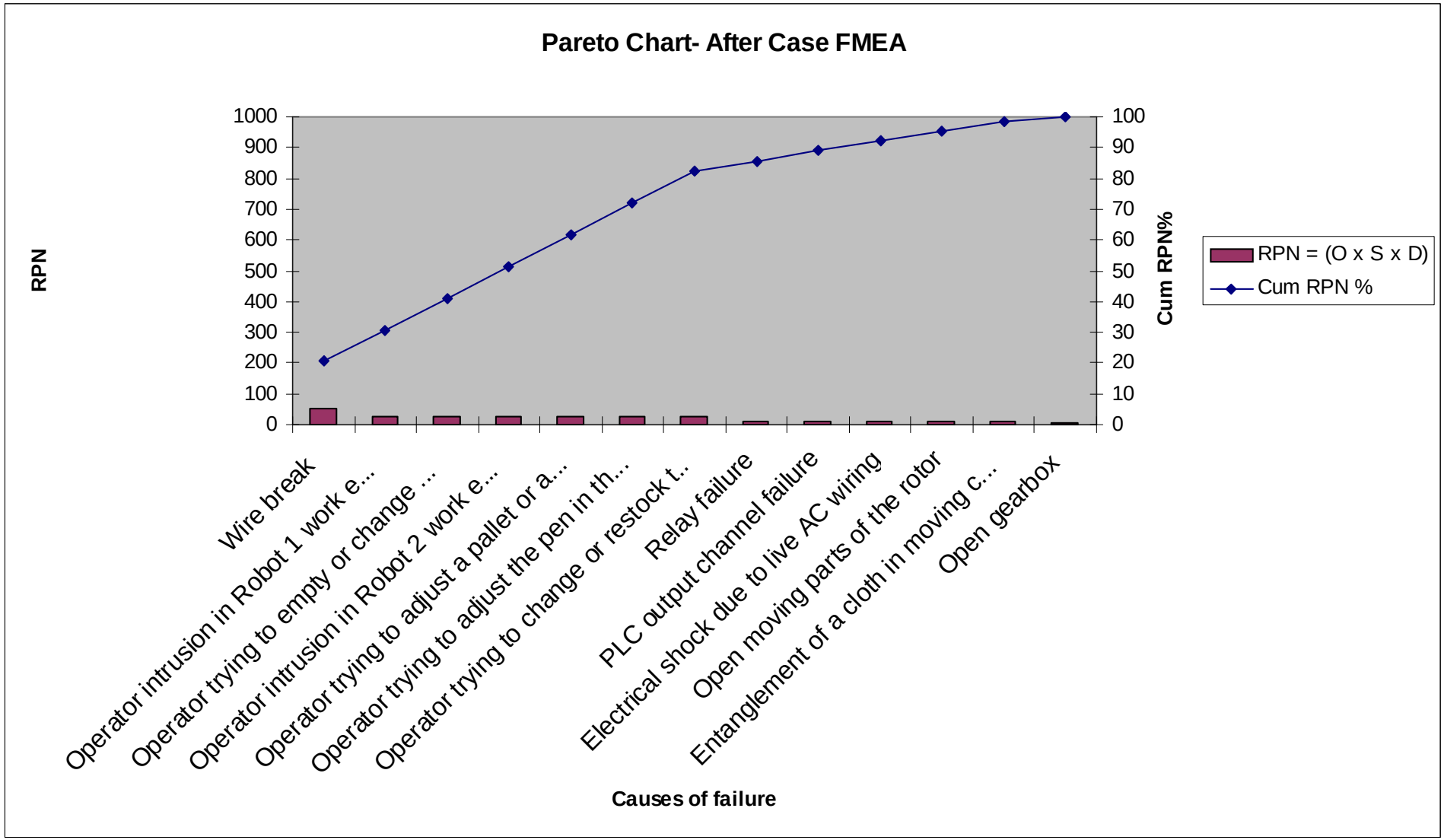


Figure 7.1: Pareto-Chart for “After Case” FMEA Analysis.

7.4 *Conclusion*

The Robotic Manufacturing Cell was set up by using non-safety PLC based control system. This system is referred as “Before Case” system. In the literature search, many safety analysis methods are discussed but there is no example to demonstrate how the results of these analyses can be used for a PLC based control system. However, one of the authors suggested that the conventional methods of safety analysis can be used for a PLC based control system. To test this idea, four test cases were identified to represent the functional safety needs of a “Before Case” system. FMEA, also known as one of the qualitative risk assessment techniques, was applied to these four cases. The results of FMEA provided different safety requirements for each test case. A Pareto-Chart was used to determine the approximate safety requirement for each cause of failure associated with every test case. The Risk-Graph analysis was used to verify the result of FMEA and Pareto-Chart. The Risk-Graph results were used to calculate the exact level of safety requirement for each test case. At the end, considering the common operational components, the SIL-3 safety integrity level requirement was decided for the total system. With this exercise, it is demonstrated how the qualitative risk assessment techniques were selected and applied on identified functional safety needs to determine the safety integrity level requirement of a PLC based control system.

In the literature published by many PLC manufacturers, it was found that these manufacturers promote totally integrated solutions for a process control and safety. This may not be the most effective solution for some applications. However, IEC 61508 standard recommends the implementation of independent process control and safety systems. Based

on this recommendation, two different PLC control systems were implemented. A non-safety PLC based system was implemented for executing normal operations and a safety PLC based system was implemented for executing safety-related functions. The IEC 61508 recommendation was used with various other factors to select the appropriate safety control system architecture. With this implementation, it was demonstrated how a customized solution is designed by using two products from different manufacturers to achieve desired level of safety.

This work includes various phases mentioned in an overall safety lifecycle of IEC 61508. It includes overall scope definition, hazard and risk analysis, overall safety requirements, safety requirement allocation, safety-related system implementation and overall safety validation. Therefore, this work can be used as a guideline to design and implement safety-related systems for any industrial application.

7.5 *Limitations*

In this thesis, only four test cases were considered to represent the functional safety needs. The scope of work is limited to four test cases due to time and hardware resource constraints. The scope can be expanded to identify and study all possible functional safety needs (test cases) and the safety-related system can be implemented for all these cases with additional resources.

Due to the discrete nature of the control signals, the Robotic Manufacturing Cell was used as an example of Machine Safety. This setup does not include any continuous processes. Hence, the Process Safety implementation could not be demonstrated in this setup.

Redundancy is an important availability factor that could have been demonstrated at various levels. Due to the limited hardware resources, implementation of redundancy at CPU, I/O and communication level could not be achieved. This implementation could have been used to demonstrate how the availability affects the safety in case of continuous processes.

The Siemens safety systems are capable of controlling large industrial applications. Due to the small laboratory setup, the Siemens technology could not be explored to its full potential. However, this thesis can be applied to the real-life manufacturing application to demonstrate the safety capabilities of Siemens “Safety Integrated” technology.

7.6 *Recommendations for Future Research*

In this work, a safety-related system is implemented for Machine-safety mode of operation. This work can be expanded by adding continuous processes to implement and analyze Process-safety mode of operation. This exercise will provide an opportunity to compare requirements of the safety system implementation for two different modes of operation.

Due to the implementation of safety PLC, diagnostic information related to each fault event can be recorded. These records contain information such as events of failure, causes of failure, and number of times the operator entered in a hazardous work area. A Manufacturing

Execution System or Enterprise Resource Planning applications (MES or ERP) can be interfaced with the safety PLC to access these records. This information can be used to calculate near misses, maintenance costs, system downtime due to each fault and total productivity of the system. This exercise will provide an opportunity to relate the safety with productivity of the manufacturing system. There is huge potential to demonstrate how safety affects the productivity in any manufacturing environment.

Appendices

Appendix 1: Summary of Risk Assessment methods based on their scope and principles

Analysis Method	Purpose, Scope	Principle
Hazard and operability study (HAZOP)	HAZOP is widely used for hazard identification in process industry in order to discover potential hazards and operability problems	HAZOP studies the potential deviations from the intended operation conditions. The studies are carried out by a multidisciplinary team, and the key words are used to guide the analysts.
Action error analysis (AEA)	AEA considers the operational, maintenance, control and supervision actions performed by a human being. The potential mistakes in individual actions are studied.	A checklist is used. The effects of each potential mistake on safety and on system performance, recognition of the mistake, and potential countermeasures are planned
Fault tree analysis (FTA)	FTA models the cause sequence leading to the TOP event. FTA can be used as a quantitative method	The causes are modeled backwards, and the probability of the TOP event is assessed on the basis of this model and reliability figures of the system components
Software fault tree analysis (SFTA)	SFTA is an extension of FTA where the TOP event is critical software fault	The software studies backwards through the program to the software inputs. SFTA attempts to prove that the program will not in any environment allow a particular unsafe output to occur.
Event tree analysis (ETA)	ETA models the sequence of potential consequences of a hazardous situation or	ETA works forward starting from hazardous situation, and the potential consequences are

	event. ETA can be used as a quantitative method	modeled
Failure mode and effect analysis (FMEA)	In FMEA the possible failures of the system components or subsystems and the consequences are analyzed systematically. FMEA is commonly used for mechanical, electrical and electronic components	The components of the system and their failures and failure modes are listed on a tabular sheet. Checklists can be used to support the analysis.
Reliability assessment	Reliability assessment means quantitative studies on potential component and equipment failures, their causes and consequences	Block diagram or FTA is used as a basis. The most important measures in reliability assessment are failure rate and time concepts.
As Low As Reasonably Practicable (ALARP)	ALARP defines the tolerable risk as that risk where additional spending on risk reduction would be in disproportion to the actually obtainable reduction of risk.	ALARP takes into account both random and systematic errors and gives emphasis not only to technical requirements, but also to the management of the safety activities for the whole safety lifecycle.

Appendix 2: IEC 61508, 7 part framework

IEC 61508 Part Number	Contents	Scope and Purpose
IEC 61508-1	General requirements	Tells us how to manage the overall safety project by using the safety life cycle approach. It uses the safety life cycle as a framework for a set of requirements to be carried out at each phase of the project.
IEC 61508-2	Requirements for electrical/ electronic/ programmable electronic safety-related systems.	Defines SIS design requirements and the detailed procedures to be observed in developing, building and testing the equipment.
IEC 61508-3	Software requirements	Details the software engineering practices that must be observed for a programmable system to qualify for safety duties. It scales the special engineering requirements against the SILs. This part is largely aimed at developers of operating systems for safety-certified components
IEC 61508-4	Definitions and abbreviations	Provides definitions of terms

IEC 61508-5	Examples of methods for the determination of safety integrity levels	Provides advice on methods of determining the SIL requirements from information obtained from hazard studies. It also defines various methods of quantitative and qualitative analysis including risk graphs.
IEC 61508-6	Guidelines on the application of IEC 61508-2 and IEC 61508-3	Provides guidance on how to carry out the requirements defined in parts 1, 2 and 3. In particular, this part contains useful sections on how to do the reliability calculations used to evaluate the SIL of a proposed design
IEC 61508-7	Overview of measures and techniques.	Provides references to further reading and techniques used in support of the SIS design work.

Appendix 3: List of the pins on the DI/ DO C2 connector and their function

Digital Inputs		**Digital Outputs**	
Pin	**Description**	**Pin**	**Description**
A	DI01 (Note 1)	e	Cycle Running (Note 6)
B	DI02 (Note 2)	f	Error (Note 7)
C	DI03	g	At home (Note 8)
D	DI04	h	Unable to move home (Note 9)
E	DI05	j	Common for e, f, g, h
F	DI06	k	Common for q, r, s, t
G	DI07	m	Common for y, z, AA, AB
H	DI08	n	Common for AC, AD, AE, AF
J	DI09	p	Common for u, v, w, x
K	DI10	q	DO01 (Note 10)
L	DI11	r	DO02 (Note 11)
M	DI12/ Command bit 0	s	DO03
N	DI13/ Command bit 1	t	DO04
O	DI14/ Command bit 2	u	DO05
P	DI15/ Command bit 3	v	DO06
R	DI16/ Command bit 4	w	DO07
S	DI Ground	x	DO08
T	DI Ground	y	DO09
U	DI Ground	z	DO10
V	DI Ground	AA	DO11/ Manipulator power
W	Inhibit move to home (Note 3)	AB	DO12/ Online
X	Emergency stop (Note 4)	AC	DO13/ Manual Mode
Y	Manipulator power (Note 5)	AD	DO14/ Cycsle stopping
Z	Manipulator power (Note 5)	AE	DO15/ Overtime
a	Not Used/ Command Strobe	AF	DO16/ Op Panel disabled
b	Not used	AG	Controller frame ground (Note 12)
c	Not used		

d	Not used		

Note 1	Can only be used for Z axis down
Note 2	Can only be used for Z axis up
Note 3	Connecting this point to DI ground will inhibit movement to home position (See note 9)
Note 4	Connecting this point to DI ground will provide an emergency stop function
Note 5	Connecting points Y and Z together will provide a manipulator power ON function
Note 6	DO point is ON during auto mode
Note 7	DO point is ON during error condition
Note 8	DO point is ON when arms at home position
Note 9	DO point is ON when return Home key is pressed and point W is connected to ground
Note 10	Can only be used for moving Z axis
Note 11	For gripper, if installed
Note 12	Controller frame ground

Appendix 4: List of 5-bits command codes and applicable function

Command Code					Function
DI #					
12	13	14	15	16	
0	1	0	0	1	Auto Mode
0	1	1	1	1	Disable operator panel
1	1	1	1	0	Enable operator panel
0	0	1	1	0	Recall memory
0	1	0	1	0	Reset Error
1	0	0	0	1	Return home
1	1	0	0	0	Select application 1
0	0	1	0	1	Select application 2
1	0	1	0	0	Select application 3
0	1	1	0	0	Select application 4
1	1	1	0	1	Select application 5
0	0	0	1	1	Start cycle
1	0	1	1	1	Step
1	1	0	1	1	Stop and memory
1	0	0	1	0	Stop cycle

Appendix 5: I/O list for PLC, Robot and sensors connection

Rack #2 Connections:

RACK#2 (ROBOT 1) DC Module				
Datafile (Octal)	**PLC Rack**	**Description**	**Type**	**External Device**
I:020/0	I:020/0	R1 Cycle Done	Robot Output	Robot DO 001 (q)
I:020/1	I:020/1	R1 Inspection Done	Robot Output	Robot DO 002 (r)
I:020/2	I:020/2	End Effector Error Robot1: Send	Robot Output	Robot DO 003 (s)
I:020/3	I:020/3			
I:020/4	I:020/4			
I:020/5	I:020/5			
I:020/6	I:020/6	Cycle Running- Robot 1	Robot Internal Status Output bit	Robot DO (e)
I:020/7	I:020/7	Error - Robot 1		Robot DO (f)
I:020/10	I:020/8	At Home - Robot 1		Robot DO (g)
I:020/11	I:020/9	Unable to move Home - Robot 1	Robot Internal Output	Robot DO (h)
I:020/12	I:020/10	Manipulator Power ON - Robot 1		Robot DO11 (AA)
I:020/13	I:020/11	Online - Robot 1		Robot DO12 (AB)
I:020/14	I:020/12	Manual Mode - Robot 1		Robot DO13 (AC)

I:020/15	I:020/13	Cycle stopping - Robot 1		Robot DO14 (AD)
I:020/16	I:020/14	Overtime - Robot 1		Robot DO15 (AE)
I:020/17	I:020/15	Op Panel Disabled - Robot 1		Robot DO16 (AF)

Datafile (Octal)	**PLC Rack**	**Description**	**Type**	**External Device**
O:020/0	O:020/0	R1 Error Clear	Robot Input	Robot DI 001(A)
O:020/1	O:020/1	R1 Start Cycle	Robot Input	Robot DI 002(B)
O:020/2	O:020/2			
O:020/3	O:020/3			
O:020/4	O:020/4			
O:020/5	O:020/5			
O:020/6	O:020/6			
O:020/7	O:020/7	Inhibit Move to Home - Robot 1	Robot Internal Input	Robot DI (W) (Grnd)
O:020/10	O:020/8	Emergency Stop - Robot 1		Robot DI (X) (Grnd)
O:020/11	O:020/9	Manipulator Power(Y&Z)- Robot 1		Robot DI (Y,Z)
O:020/12	O:020/10	Command Strobe - Robot 1	Robot Internal Command Input bits	Robot DI (a)
O:020/13	O:020/11	Command Bit 0 - Robot 1		Robot DI 012 (M)
O:020/14	O:020/12	Command Bit 1 - Robot 1		Robot DI 013 (N)

O:020/15	O:020/13	Command Bit 2 - Robot 1		Robot DI 014 (O)
O:020/16	O:020/14	Command Bit 3 - Robot 1		Robot DI 015 (P)
O:020/17	O:020/15	Command Bit 4 - Robot 1		Robot DI 016 (R)
End Effector Robot1			Robot D1 003 (C)	
Input Ground Robot 1			Robot DI (S, T, U, V)	
Output Ground Robot 1			Robot DO (j, k, l, m, n, p)	
Controller Frame Ground Robot 1			Robot (AG)	

Rack #4 Connections

RACK#4 (ROBOT 2) DC Module				
Datafile (Octal)	**PLC Rack**	**Description**	**Type**	**External Device**
I:040/0	I:040/0	R2 Cycle Done	Robot Output	Robot DO 001 (q)
I:040/1	I:040/1	Pen picked (Done)	Robot Output	Robot DO 002 (r)
I:040/2	I:040/2	Char Done	Robot Output	Robot DO 003 (s)
I:040/3	I:040/3	End Effector Error Robot2: Send	Robot Output	Robot DO 004 (t)
I:040/4	I:040/4			
I:040/5	I:040/5			

I:040/6	I:040/6	Cycle Running - Robot 2	Robot Internal Status Output bit	Robot DO (e)
I:040/7	I:040/7	Error- Robot 2		Robot DO (f)
I:040/10	I:040/8	At Home- Robot 2		Robot DO (g)
I:040/11	I:040/9	Unable to move Home -Robot 2	Robot Internal Output	Robot DO (h)
I:040/12	I:040/10	Manipulator power ON- Robot 2		Robot DO 011 (AA)
I:040/13	I:040/11	Online- Robot 2		Robot DO 012 (AB)
I:040/14	I:040/12	Manual Mode- Robot 2		Robot DO 013 (AC)
I:040/15	I:040/13	Cycle stopping- Robot 2		Robot DO 014 (AD)
I:040/16	I:040/14	Overtime- Robot 2		Robot DO 015 (AE)
I:040/17	I:040/15	Op Panel Disabled- Robot 2		Robot DO 016 (AF)

Datafile (Octal)	**PLC Rack**	**Description**	**Type**	**External Device**
O:040/0	O:040/0	Char bit 0/ Pen select 1	Robot Input	Robot DI 001 (A)
O:040/1	O:040/1	Char bit 1/Pen select 2	Robot Input	Robot DI 002 (B)
O:040/2	O:040/2	Char bit 2	Robot Input	Robot DI 003 (C)
O:040/3	O:040/3	Char bit 3	Robot Input	Robot DI 004 (D)
O:040/4	O:040/4	Char bit 4	Robot Input	Robot DI 005 (E)
O:040/5	O:040/5	Read Next Char	Robot Input	Robot DI 006 (F)

O:040/6	O:040/6	R2 Error Clear: end effector	Robot Input	Robot DI 009 (I)
O:040/7	O:040/7	R2 Start Cycle/Pen check done	Robot Input	Robot DI 007 (G)
O:040/10	O:040/8	Emergency Stop - Robot 2	Robot Internal Input	Robot DI (X) (Grnd)
O:040/11	O:040/9	Manipulator Power(Y&Z) - Robot 2		Robot DI (Y,Z)
O:040/12	O:040/10	Command Strobe - Robot 2	Robot Internal Command Input bits	Robot DI (a)
O:040/13	O:040/11	Command Bit 0 - Robot 2		Robot DI 012 (M)
O:040/14	O:040/12	Command Bit 1 - Robot 2		Robot DI 013 (N)
O:040/15	O:040/13	Command Bit 2 - Robot 2		Robot DI 014 (O)
O:040/16	O:040/14	Command Bit 3 - Robot 2		Robot DI 015 (P)
O:040/17	O:040/15	Command Bit 4 - Robot 2		Robot DI 016 (R)
End Effector Robot 2			Robot DI 008 (H)	
Input Ground Robot 2			Robot DI (S, T, U, V)	
Output Ground Robot 2			Robot DO (j, k, m, n, p)	
Controller Frame Ground Robot 2			Robot (AG)	

Rack #1 Connections

RACK#1 (ROBOT 3) DC Module				
Datafile (Octal)	**PLC Rack**	**Description**	**Type**	**External Device**
I:010/0	I:010/0	R3 Cycle Done	Robot Output	Robot DO 001 (q)
I:010/1	I:010/1	**********Not used**********		
I:010/2	I:010/2	End Effector Error Robot 3	Robot Output	Robot DO 002 (r)
I:010/3	I:010/3			
I:010/4	I:010/4			
I:010/5	I:010/5			
I:010/6	I:010/6	Cycle running- Robot 3	Robot Internal Status Output bit	Robot DO (e)
I:010/7	I:010/7	Error- Robot 3		Robot DO (f)
I:010/10	I:010/8	At home- Robot 3		Robot DO (g)
I:010/11	I:010/9	Unable to move Home -Robot 3	Robot Internal Output	Robot DO (h)
I:010/12	I:010/10	Manipulator power ON- Robot 3		Robot DO 011 (AA)
I:010/13	I:010/11	Online- Robot 3		Robot DO 012 (AB)
I:010/14	I:010/12	Manual Mode- Robot 3		Robot DO 013 (AC)
I:010/15	I:010/13	Cycle stopping- Robot 3		Robot DO 014 (AD)

I:010/16	I:010/14	Overtime- Robot 3		Robot DO 015 (AE)
I:060/17	I:010/15	Op Panel Disabled- Robot 3		Robot DO 016 (AF)

Datafile (Octal)	PLC Rack	Description	Type	External Device
O:010/0	O:010/0	Start Cycle to Robot 3	Robot Input	Robot DI 001 (A)
O:010/1	O:010/1	Error Clear : Robot 3	Robot Input	Robot DI 002 (B)
O:010/2	O:010/2	Bad from Vision system (1/0)	Camera output	Robot DI 003 (C)
O:010/3	O:010/3	Good from Vision system (1/0)	Camera output	Robot DI 003 (D)
O:010/4	O:010/4			
O:010/5	O:010/5			
O:010/6	O:010/6	Inhibit Move to Home - Robot 2	Robot Internal Input	Robot DI (W) (Grnd)
O:010/7	O:010/7	Inhibit Move to Home - Robot 3		Robot DI (W) (Grnd)
O:010/10	O:010/8	Emergency Stop - Robot 3		Robot DI (X) (Grnd)
O:010/11	O:010/9	Manipulator Power(Y&Z) - Robot 3		Robot DI (Y,Z)
O:010/12	O:010/10	Command Strobe - Robot 3	Robot Internal Command Input bits	Robot DI (a)
O:010/13	O:010/11	Command Bit 0 - Robot 3		Robot DI 012 (M)
O:010/14	O:010/12	Command Bit 1 - Robot 3		Robot DI 013 (N)
O:010/15	O:010/13	Command Bit 2 - Robot 3		Robot DI 014 (O)

O:010/16	O:010/14	Command Bit 3 - Robot 3		Robot DI 015 (P)
O:060/17	O:010/15	Command Bit 4 - Robot 3		Robot DI 016 (R)
End Effector Robot 3			Robot D1 004 (D)	
Input Ground Robot 3			Robot DI (S, T, U, V)	
Output Ground Robot 3			Robot DO (j, k, m, n, p)	
Controller Frame Ground Robot 3			Robot (AG)	

Rack #6 Connections

RACK#6 (Conveyor) DC Module				
Datafile (Octal)	**PLC Rack**	**Description**	**Type**	**External Device**
I:060/0	I:060/0	Pallet Present @ Station 0	On Conveyor	Proximity sensor
I:060/1	I:060/1	Pallet Present @ Station 1		
I:060/2	I:060/2	Pallet Present @ Station 2		
I:060/3	I:060/3	Pallet Present @ Station 3		
I:060/4	I:060/4	Pallet Present @ Station 4		
I:060/5	I:060/5	Pallet placed correctly		
I:060/6	I:060/6	Pallet present before st0		Light beam sensor
I:060/7	I:060/7	Paper Stack Empty	On Paper Feeder	

I:060/10	I:060/8	Pen present 1 @ feeder 2	On Pen Feeder	
I:060/11	I:060/9	Pen present 2 @ feeder 2		
I:060/12	I:060/10	Pen present 3 @ feeder 2		
I:060/13	I:060/11	output 1: pass		Vision System
I:060/14	I:060/12	output 2: busy		
I:060/15	I:060/13			
I:060/16	I:060/14			
I:060/17	I:060/15			

Datafile (Octal)	**PLC Rack**	**Description**	**Type**	**External Device**
O:060/0	O:060/0	Motor Running	Relay Logic	motor
O:060/1	O:060/1	Trigger		
O:060/2	O:060/2			
O:060/3	O:060/3			
O:060/4	O:060/4			
O:060/5	O:060/5			
O:060/6	O:060/6			
O:060/7	O:060/7			

O:060/10	O:060/8			
O:060/11	O:060/9			
O:060/12	O:060/10			
O:060/13	O:060/11			
O:060/14	O:060/12			
O:060/15	O:060/13			
O:060/16	O:060/14			
O:060/17	O:060/15			

Rack #3 Connections

RACK#3 (Conveyor) AC Module				
Datafile	**PLC Rack**	**Description**	**Type**	**External Device**
I:030/0	SAME as OCTAL	Motor Input	relay logic	From the start and stop switches
I:030/1				
I:030/2				
I:030/3				
I:030/4				
I:030/5				

I:030/6				
I:030/7				
I:030/10				
I:030/11				
I:030/12				
I:030/13				
I:030/14				
I:030/15				
I:030/16				
I:030/17				

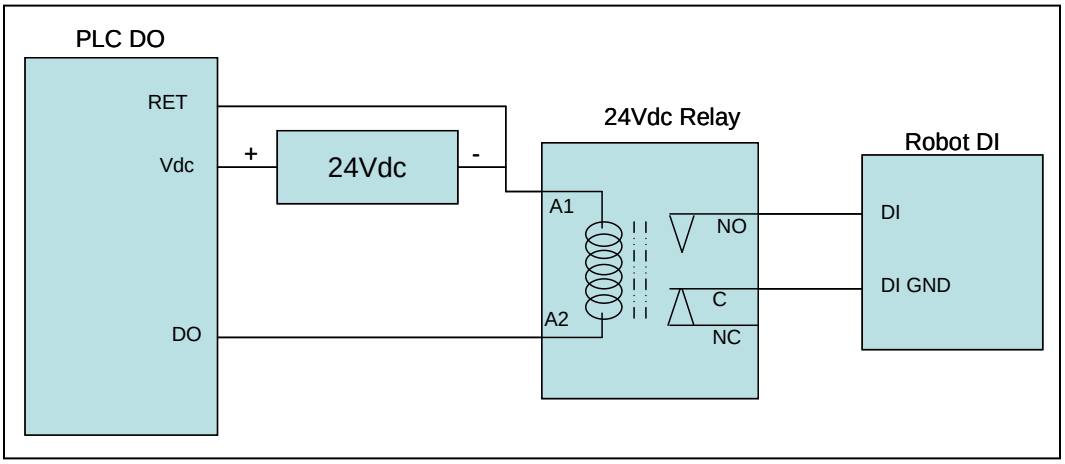
Datafile	PLC Rack	Description	Type	External Device
O:030/0	SAME as OCTAL	Stopper at Station0	Pneumatics	On Conveyor
O:030/1		Stopper at Station1		
O:030/2		Stopper at Station2		
O:030/3		Stopper at Station3		
O:030/4		Stopper at Station4		
O:030/5		Clamp at Station1		
O:030/6		Clamp at Station2		
O:030/7		Pneumatic Clamp at Station3		

O:030/10				
O:030/11				
O:030/12				
O:030/13				
O:030/14				
O:030/15				
O:030/16				
O:030/17				

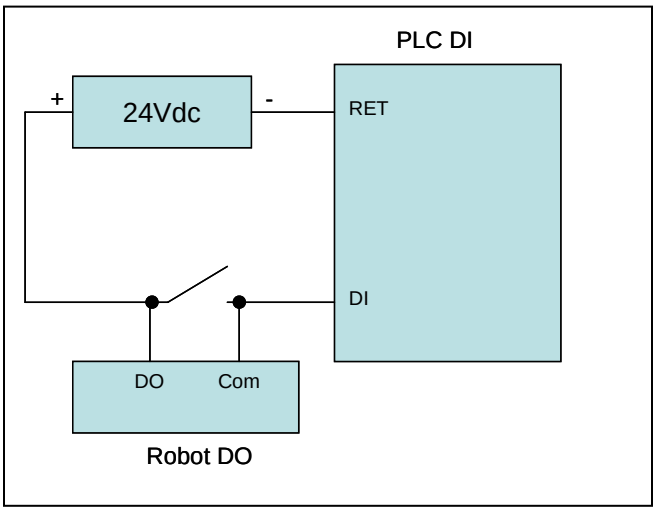
Appendix 6: The wiring schematic for the connection between Robot I/Os and PLC I/Os

Robot DI/ DO Integration with PLC:

Connection from PLC DO to Robot DI:



Connection from Robot DO to PLC DI:



Appendix 7: List of the sensors and actuators and their location

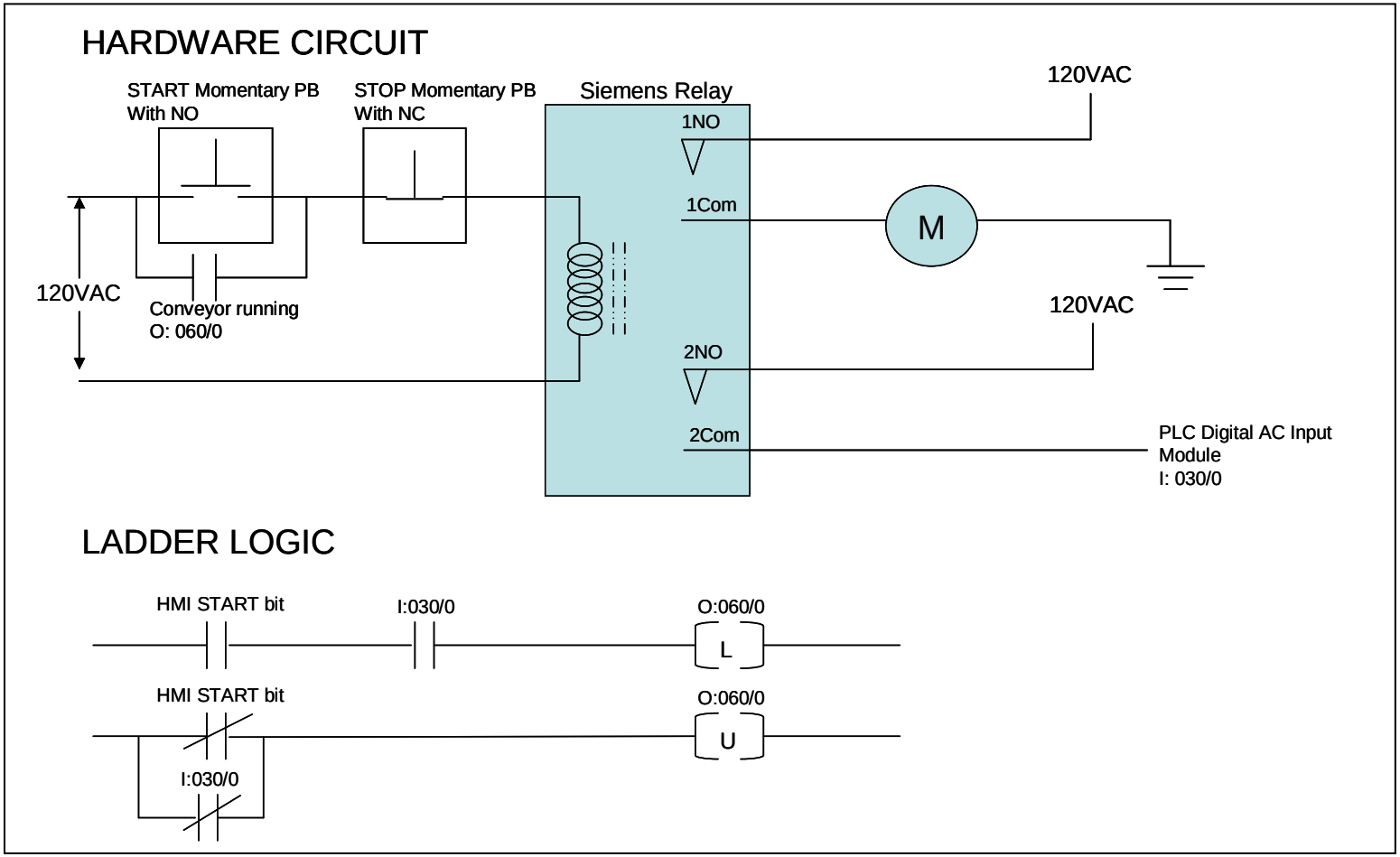
Sensors:

Sr. No.	**Sensor Input**	**Type of sensor**	**PLC Interface**	**Make**
1	Pallet placed correctly before station 0	Proximity Sensor	DI	Banner Engineering
2	Pallet Present @ Station 0	Proximity Sensor	DI	Banner Engineering
3	Pallet Present @ Station 1	Proximity Sensor	DI	Banner Engineering
4	Pallet Present @ Station 2	Proximity Sensor	DI	Banner Engineering
5	Pallet Present @ Station 3	Proximity Sensor	DI	Banner Engineering
6	Pallet Present @ Station 4	Proximity Sensor	DI	Banner Engineering
7	Paper Stack Empty	Photo Beam Sensor	DI	Banner Engineering
8	Pen 1 present @ Pen Feeder	Photo Beam Sensor	DI	Banner Engineering
9	Pen 2 present @ Pen Feeder	Photo Beam Sensor	DI	Banner Engineering
10	Pen 3 present @ Pen Feeder	Photo Beam Sensor	DI	Banner Engineering
11	START	Momentary Push Button- Green	Hardwired	Allen-Bradley
12	STOP	Momentary Push Button- Red	Hardwired	Allen-Bradley

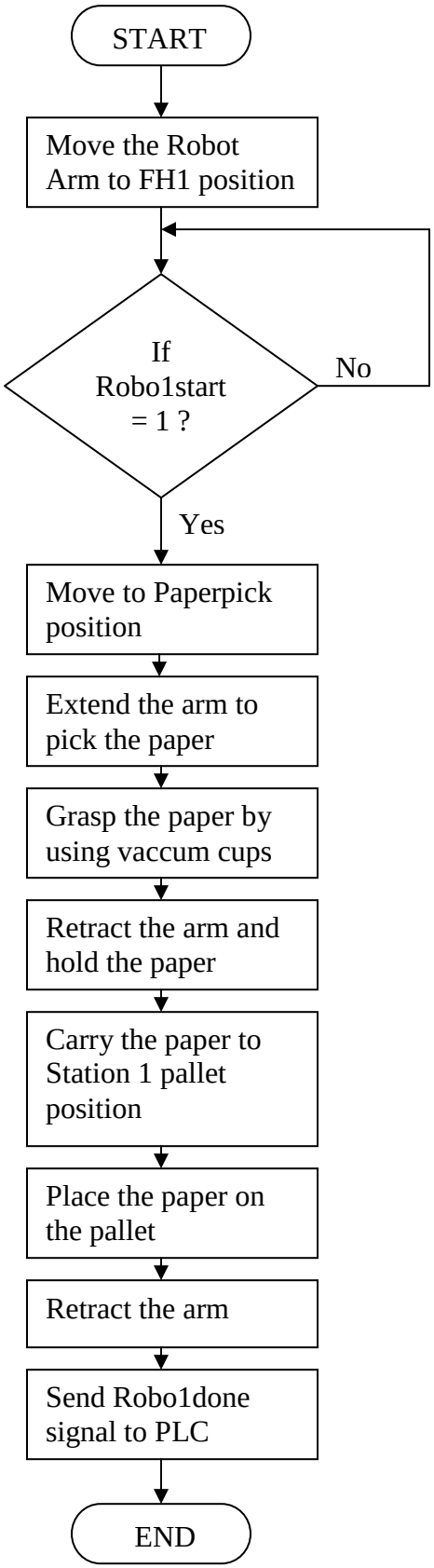
Actuators:

Sr. No.	Actuator Output	Type of sensor	PLC Interface	Make
1	Stopper at Station0	Pneumatic Solenoid Relay	DO	SMC
2	Stopper at Station1	Pneumatic Solenoid Relay	DO	SMC
3	Stopper at Station2	Pneumatic Solenoid Relay	DO	SMC
4	Stopper at Station3	Pneumatic Solenoid Relay	DO	SMC
5	Stopper at Station4	Pneumatic Solenoid Relay	DO	SMC
6	Clamp at Station1	Pneumatic Solenoid Relay	DO	SMC
7	Clamp at Station2	Pneumatic Solenoid Relay	DO	SMC
8	Clamp at Station3	Pneumatic Solenoid Relay	DO	SMC
9	Conveyor	Motor	Hardwired	Leeson

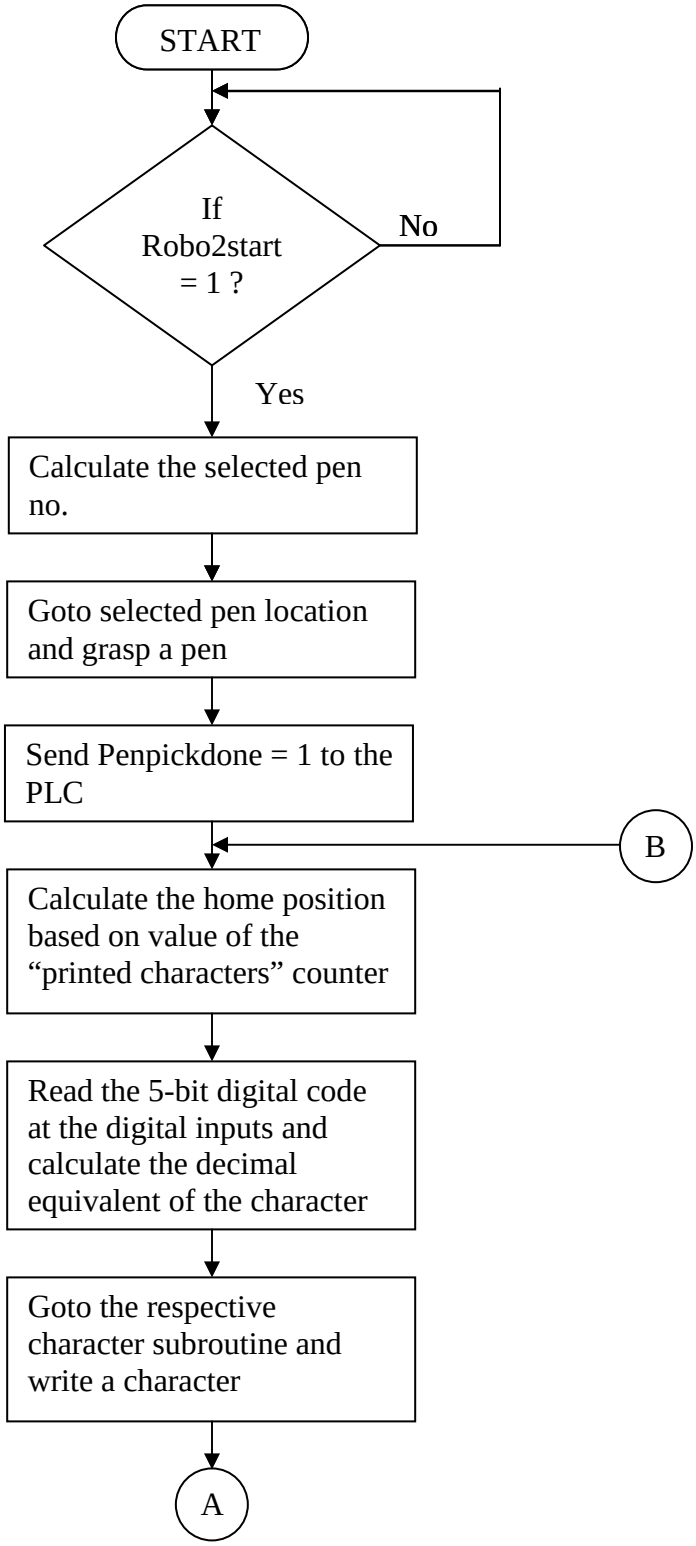
Appendix 8: The wiring schematic of the motor control operation

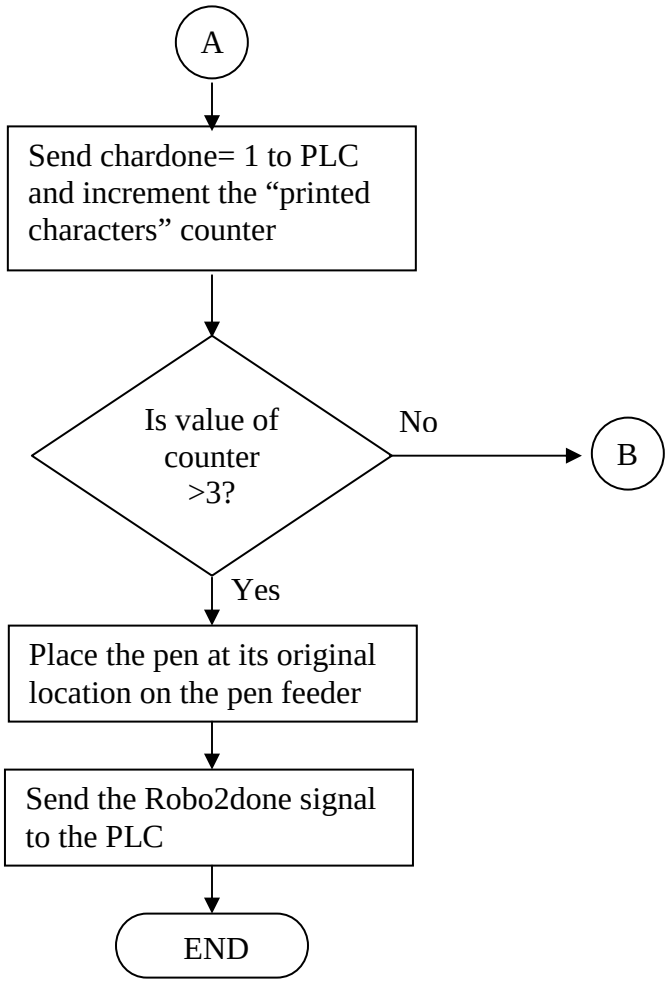


Appendix 9: Flow chart for Robot 1



Appendix 10: Flow chart for Robot 2




A
Send chardone= 1 to PLC and increment the “printed characters” counter
Is value of counter >3?
No
B
Yes
Place the pen at its original location on the pen feeder
Send the Robo2done signal to the PLC
END

Appendix 11: Robot 1 AML program

```
--** POINTS DECLARED **--
FH1: NEW PT(12,22,0,0);
PAPERPICK: NEW PT(10,22,0,0);
DOWN: NEW -5;
UP: NEW 0;
DROP1: NEW PT(-10,20,0,0);
---------***** DI DECLARED ******----------------------
ROBO1START: NEW 2;  --INPUT--
ERRORCLEAR: NEW 1;  --INPUT--
---------***** DO DECLARED ******----------------------
ROBO1DONE: NEW 1;   --OUTPUT--
ROBO1INSPECDONE: NEW 2; --OUTPUT--
SENDERROR: NEW 3;   --OUTPUT--
--------****END OF DECLARATION AND START OF MAIN PROGRAM****--------
MAIN: SUBR;
   WRITEO(ROBO1DONE,0);
   PMOVE(FH1);
  LABEL2:
   TESTI(ROBO1START,1,LABEL1);
   BRANCH(LABEL2);
  LABEL1:
   PMOVE(PAPERPICK);
   ZMOVE(DOWN);
   GRASP;                    --PAPER GRASPED FROM THE FEEDER STACK--
   ZMOVE(UP);
   PMOVE(DROP1);
   PAYLOAD(1);
   ZMOVE(DOWN);
   RELEASE;                      -- PAPER PLACED ON THE PALLET--
   ZMOVE(UP);
   WRITEO(ROBO1DONE,1);
   END;
```

Appendix 12: Robot 2 AML program

```
----------****** CONSTANTS DECLARED ******----------
 RESULT: STATIC COUNTER;
 CHARNUM: STATIC COUNTER;
 PENRESULT: STATIC COUNTER;
 PENDOWNDONE: STATIC COUNTER;
 PT1: NEW 0.2;
 PT2: NEW 0.4;
 PT3: NEW 0.6;
 PT4: NEW 0.8;
 PT5: NEW 1.0;
 PT6: NEW 1.2;
 UP: NEW 0;
 DOWN: NEW -0.327;
 PENDOWN: NEW -3.841;
 ----------****** POINTS DECLARED ******----------
 HP1: NEW PT(-1.9,23.5,0,-60);
 HP2: NEW PT(-0.2,23.450,0,-60);
 HP3: NEW PT(1.1,23.450,0,-60);
 FH2: NEW PT(20,12,0,-60);
 PICKPEN1: NEW PT(10,16.5,0,-60);
 PICKPEN2: NEW PT(13,16.520,0,-60);
 PICKPEN3: NEW PT(16.750,16.60,0,-60);
----------****  DI DECLARED****----------
 NEXTCHAR: NEW 6;              --DI006(F) INITIALIZED--
 ROBO2START: NEW 7;            --DI007(G) INITIALIZED--
 ENDEFFECTOR: NEW 8;           --DI008(H) INITIALIZED--
 ERRORCLEAR: NEW 9;            --DI009(I) INITIALIZED--
----------**** DO DECLARED****----------
 ROBO2DONE: NEW 1;             --DO001(Q) INITIALIZED--
 PENPICKDONE: NEW 2;           --DO002(R) INITIALIZED--
 CHARDONE: NEW 3;              --DO003(S) INITIALIZED--
 SENDERROR: NEW 4;             --DO004(T) INITIALIZED--
----------****** MAIN PROGRAM ******----------
 MAIN: SUBR;
----------****** TESTDIS SUBROUTINE ****** ----------
 TESTDIS:SUBR(FIRST,LAST);
     SETC(RESULT,0);
   LOOP:
     COMPC(FIRST LT LAST, DONE);
     SETC(RESULT,2*RESULT+TESTI(FIRST));
     DECR(FIRST);
     BRANCH(LOOP);
  DONE:
```

```
      END;
----------****** NUM SUBROUTINE ******----------
 NUM: SUBR;
    TESTC(CHARNUM,1,HOME1);
    TESTC(CHARNUM,2,HOME2);
    TESTC(CHARNUM,3,HOME3);
   HOME1:
    PMOVE(HP1);
    BRANCH(DONE1);
   HOME2:
    PMOVE(HP2);
    BRANCH(DONE1);
   HOME3:
    PMOVE(HP3);
    BRANCH(DONE1);
   DONE1:
    END;
----------*****PICK PEN SUBROUTINE ***** ----------

 PICKPEN:SUBR(ONE,TWO);
    SETC(PENRESULT,0);
   LOOP:
    COMPC(ONE LT TWO, DONE);
    SETC(PENRESULT,2*PENRESULT+TESTI(ONE));
    DECR(ONE);
    BRANCH(LOOP);
  DONE:
     END;
----------****PEN DROP SUBROUTINE****----------
 PENDROP: SUBR;
   TESTC(PENRESULT,1,PEN1DROP);
   TESTC(PENRESULT,2,PEN2DROP);
   TESTC(PENRESULT,3,PEN3DROP);
 PEN1DROP:
   PMOVE(PICKPEN1);
   DELAY(1);
   ZMOVE(PENDOWN);
   DELAY(1);
   RELEASE;
   DELAY(1);
   ZMOVE(UP);
   DELAY(1);
   PMOVE(FH2);
   BRANCH(ROBODONE);
 PEN2DROP:
   PMOVE(PICKPEN2);
```

```
   DELAY(1);
   ZMOVE(PENDOWN);
   DELAY(1);
   RELEASE;
   DELAY(1);
   ZMOVE(UP);
   DELAY(1);
   PMOVE(FH2);
   BRANCH(ROBODONE);
 PEN3DROP:
   PMOVE(PICKPEN3);
   DELAY(1);
   ZMOVE(PENDOWN);
   DELAY(1);
   RELEASE;
   DELAY(1);
   WRITEO(PENPICKDONE,0);
   ZMOVE(UP);
   DELAY(1);
   PMOVE(FH2);
   BRANCH(ROBODONE);
 ROBODONE:
   END;

-----****END OF SUBROUTINE AND START OF MAIN PRGM****----------

 START:
  WRITEO(CHARDONE,0);
  WRITEO(PENPICKDONE,0);
  WRITEO(ROBO2DONE,0);
  TESTI(ROBO2START,1,PENSELECT);
  BRANCH(START);
  PENSELECT:
  PICKPEN(2,1);
  TESTC(PENRESULT,1,P1);
  TESTC(PENRESULT,2,P2);
  TESTC(PENRESULT,3,P3);
  BRANCH(PENSELECT);
 P1:
   PMOVE(PICKPEN1);
   ZMOVE(PENDOWN);
   BRANCH(PENPICKED);
 P2:
   PMOVE(PICKPEN2);
   ZMOVE(PENDOWN);
   BRANCH(PENPICKED);
```

```
P3:
  PMOVE(PICKPEN3);
  ZMOVE(PENDOWN);
  BRANCH(PENPICKED);
PENPICKED:
  DELAY(1);
  GRASP;
  DELAY(1);
  ZMOVE(UP);
  WRITEO(PENPICKDONE,1);
  PMOVE(HP1);
  SETC(CHARNUM,1);
  LINEAR(1);
LABEL3:
 TESTI(NEXTCHAR,1,LABEL1);
 BRANCH(LABEL3);
LABEL1:
 NUM;
LABEL2:
 TESTDIS(5,1);
 WRITEO(CHARDONE,1);
 TESTC(RESULT,0,LABEL2);
 TESTC(RESULT,1,A);
 TESTC(RESULT,2,B);
 TESTC(RESULT,3,C);
 TESTC(RESULT,4,D);
 TESTC(RESULT,5,E);
 TESTC(RESULT,6,F);
 TESTC(RESULT,7,G);
 TESTC(RESULT,8,H);
 TESTC(RESULT,9,I);
 TESTC(RESULT,10,J);
 TESTC(RESULT,11,K);
 TESTC(RESULT,12,L);
 TESTC(RESULT,13,M);
 TESTC(RESULT,14,N);
 TESTC(RESULT,15,O);
 TESTC(RESULT,16,P);
 TESTC(RESULT,17,Q);
 TESTC(RESULT,18,R);
 TESTC(RESULT,19,S);
 TESTC(RESULT,20,T);
 TESTC(RESULT,21,U);
 TESTC(RESULT,22,V);
 TESTC(RESULT,23,W);
 TESTC(RESULT,24,X);
```

```
 TESTC(RESULT,25,Y);
 TESTC(RESULT,26,Z);
DONE2:
  WRITEO(CHARDONE,0);
  INCR(CHARNUM);
  COMPC(CHARNUM <=3,LABEL3);
  WRITEO(ROBO2DONE,1);
  LINEAR(0);
  PENDROP;
  WRITEO(PENPICKDONE,0);
  BRANCH(OVER);
 A:
    ZMOVE(DOWN);
    DPMOVE (<PT2,PT6,0,0>);
    DPMOVE (<PT2,-PT6,0,0>);
    ZMOVE(UP);
    DPMOVE (<-PT3,PT3,0,0>);
    ZMOVE(DOWN);
    DPMOVE(<PT2,0,0,0>);
    ZMOVE(UP);
    BRANCH(DONE2);
 B:
    ZMOVE(DOWN);
    DPMOVE(<0,PT6,0,0>);
    DPMOVE(<PT3,0,0,0>);
    DPMOVE(<PT1,-PT1,0,0>);
    DPMOVE(<0,-PT1,0,0>);
    DPMOVE(<-PT1,-PT1,0,0>);
    DPMOVE(<PT1,-PT1,0,0>);
    DPMOVE(<0,-PT1,0,0>);
    DPMOVE(<-PT1,-PT1,0,0>);
    DPMOVE(<-PT3,0,0,0>);
    ZMOVE(UP);
    DPMOVE(<0,PT3,0,0>);
    ZMOVE(DOWN);
    DPMOVE(<PT3,0,0,0>);
    ZMOVE(UP);
    BRANCH(DONE2);
 C:
    DPMOVE(<PT4,PT1,0,0>);
    ZMOVE(DOWN);
    DPMOVE(<-PT1,-PT1,0,0>);
    DPMOVE(<-PT2,0,0,0>);
    DPMOVE(<-PT1,PT1,0,0>);
    DPMOVE(<0,PT4,0,0>);
    DPMOVE(<PT1,PT1,0,0>);
```

```
   DPMOVE(<PT2,0,0,0>);
   DPMOVE(<PT1,-PT1,0,0>);
   ZMOVE(UP);
   BRANCH(DONE2);
D:
   ZMOVE(DOWN);
   DPMOVE(<0,PT6,0,0>);
   DPMOVE(<PT3,0,0,0>);
   DPMOVE(<PT1,-PT1,0,0>);
   DPMOVE(<0,-PT4,0,0>);
   DPMOVE(<-PT1,-PT1,0,0>);
   DPMOVE(<-PT3,0,0,0>);
   ZMOVE(UP);
   BRANCH(DONE2);
E:
   DPMOVE(<PT4,0,0,0>);
   ZMOVE(DOWN);
   DPMOVE(<-PT4,0,0,0>);
   DPMOVE(<0,PT6,0,0>);
   DPMOVE(<PT4,0,0,0>);
   ZMOVE(UP);
   DPMOVE(<-PT4,-PT3,0,0>);
   ZMOVE(DOWN);
   DPMOVE(<PT3,0,0,0>);
   ZMOVE(UP);
   BRANCH(DONE2);
F:
   ZMOVE(DOWN);
   DPMOVE(<0,PT6,0,0>);
   DPMOVE(<PT4,0,0,0>);
   ZMOVE(UP);
   DPMOVE(<-PT4,-PT3,0,0>);
   ZMOVE(DOWN);
   DPMOVE(<PT3,0,0,0>);
   ZMOVE(UP);
   BRANCH(DONE2);
G:
   DPMOVE(<PT4,PT5,0,0>);
   ZMOVE(DOWN);
   DPMOVE(<-PT1,PT1,0,0>);
   DPMOVE(<-PT2,0,0,0>);
   DPMOVE(<-PT1,-PT1,0,0>);
   DPMOVE(<0,-PT4,0,0>);
   DPMOVE(<PT1,-PT1,0,0>);
   DPMOVE(<PT2,0,0,0>);
   DPMOVE(<PT1,PT1,0,0>);
```

```
   DPMOVE(<0,PT2,0,0>);
   DPMOVE(<-PT1,0,0,0>);
   ZMOVE(UP);
   BRANCH(DONE2);
H:
   ZMOVE(DOWN);
   DPMOVE(<0,PT6,0,0>);
   ZMOVE(UP);
   DPMOVE(<PT4,0,0,0>);
   ZMOVE(DOWN);
   DPMOVE(<0,-PT6,0,0>);
   ZMOVE(UP);
   DPMOVE(<0,PT3,0,0>);
   ZMOVE(DOWN);
   DPMOVE(<-PT4,0,0,0>);
   ZMOVE(UP);
   BRANCH(DONE2);
I:
   ZMOVE(DOWN);
   DPMOVE(<PT4,0,0,0>);
   ZMOVE(UP);
   DPMOVE(<0,PT6,0,0>);
   ZMOVE(DOWN);
   DPMOVE(<-PT4,0,0,0>);
   ZMOVE(UP);
   DPMOVE(<PT2,0,0,0>);
   ZMOVE(DOWN);
   DPMOVE(<0,-PT6,0,0>);
   ZMOVE(UP);
   BRANCH(DONE2);
J:
   DPMOVE(<0,PT6,0,0>);
   ZMOVE(DOWN);
   DPMOVE(<PT4,0,0,0>);
   ZMOVE(UP);
   DPMOVE(<-PT1,0,0,0>);
   ZMOVE(DOWN);
   DPMOVE(<0,-PT5,0,0>);
   DPMOVE(<-PT1,-PT1,0,0>);
   DPMOVE(<-PT1,0,0,0>);
   DPMOVE(<-PT1,PT1,0,0>);
   DPMOVE(<0,PT1,0,0>);
   ZMOVE(UP);
   BRANCH(DONE2);
K:
   ZMOVE(DOWN);
```

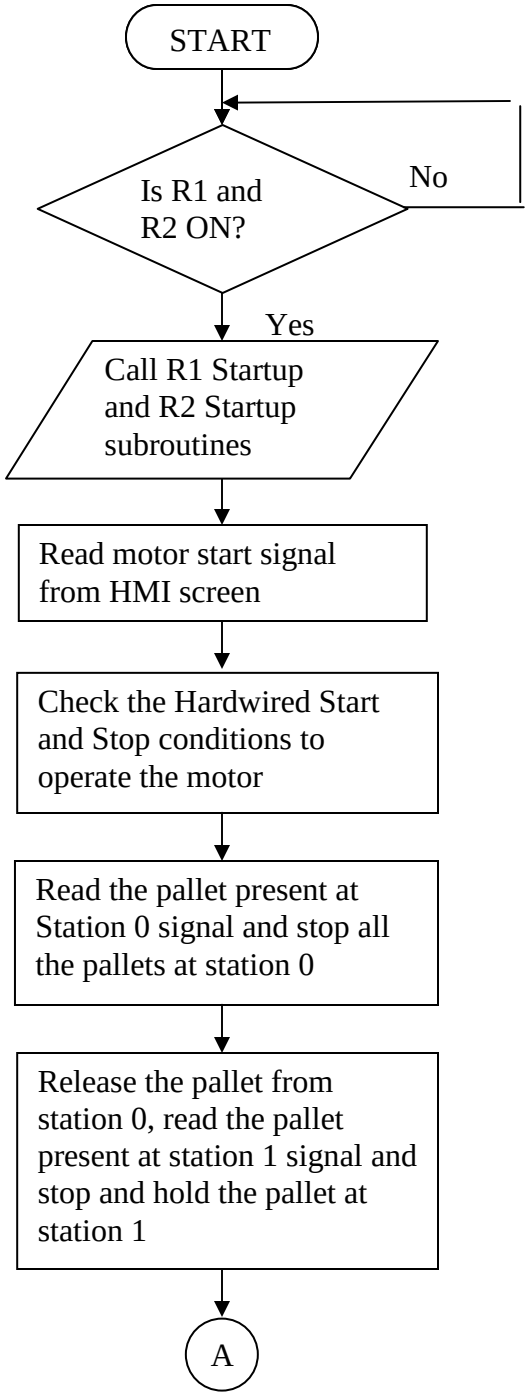
```
    DPMOVE(<0,PT6,0,0>);
    ZMOVE(UP);
    DPMOVE(<0,-PT3,0,0>);
    ZMOVE(DOWN);
    DPMOVE(<PT4,PT3,0,0>);
    ZMOVE(UP);
    DPMOVE(<-PT4,-PT3,0,0>);
    ZMOVE(DOWN);
    DPMOVE(<PT4,-PT3,0,0>);
    ZMOVE(UP);
    BRANCH(DONE2);
L:
    DPMOVE(<0,PT6,0,0>);
    ZMOVE(DOWN);
    DPMOVE(<0,-PT6,0,0>);
    DPMOVE(<PT4,0,0,0>);
    ZMOVE(UP);
    BRANCH(DONE2);
M:
    ZMOVE(DOWN);
    DPMOVE(<0,PT6,0,0>);
    DPMOVE(<PT2,-PT2,0,0>);
    DPMOVE(<PT2,PT2,0,0>);
    DPMOVE(<0,-PT6,0,0>);
    ZMOVE(UP);
    BRANCH(DONE2);
N:
    ZMOVE(DOWN);
    DPMOVE(<0,PT6,0,0>);
    DPMOVE(<PT4,-PT6,0,0>);
    DPMOVE(<0,PT6,0,0>);
    ZMOVE(UP);
    BRANCH(DONE2);
O:
    DPMOVE(<PT3,0,0,0>);
    ZMOVE(DOWN);
    DPMOVE(<-PT2,0,0,0>);
    DPMOVE(<-PT1,PT1,0,0>);
    DPMOVE(<0,PT4,0,0>);
    DPMOVE(<PT1,PT1,0,0>);
    DPMOVE(<PT2,0,0,0>);
    DPMOVE(<PT1,-PT1,0,0>);
    DPMOVE(<0,-PT4,0,0>);
    DPMOVE(<-PT1,-PT1,0,0>);
    ZMOVE(UP);
    BRANCH(DONE2);
```

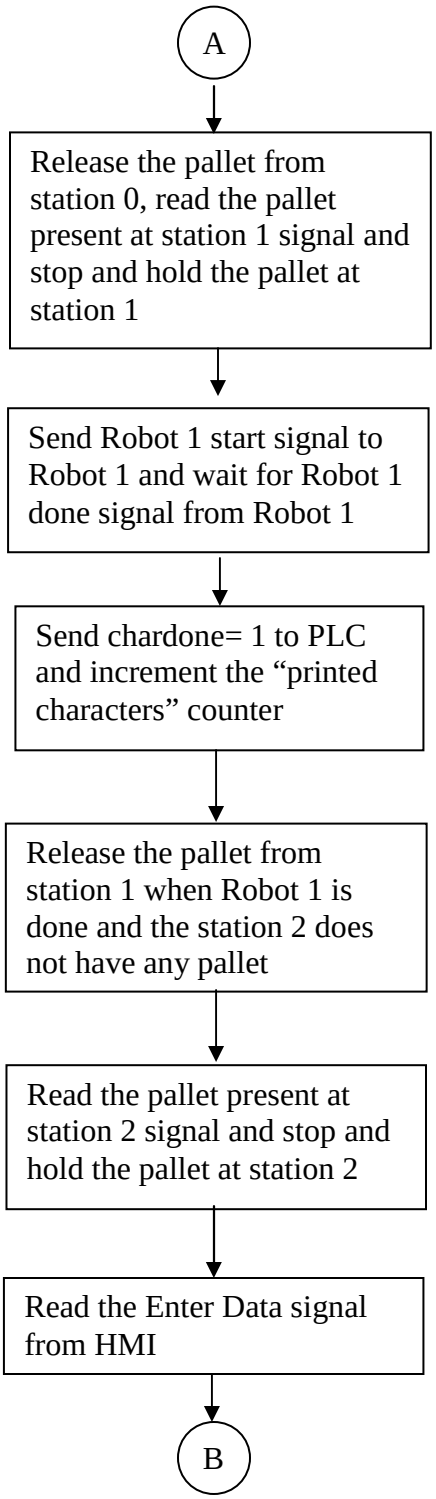
```
P:
   ZMOVE(DOWN);
   DPMOVE(<0,PT6,0,0>);
   DPMOVE(<PT3,0,0,0>);
   DPMOVE(<PT1,-PT1,0,0>);
   DPMOVE(<0,-PT1,0,0>);
   DPMOVE(<-PT1,-PT1,0,0>);
   DPMOVE(<-PT3,0,0,0>);
   ZMOVE(UP);
   BRANCH(DONE2);
Q:
   DPMOVE(<PT3,0,0,0>);
   ZMOVE(DOWN);
   DPMOVE(<-PT2,0,0,0>);
   DPMOVE(<-PT1,PT1,0,0>);
   DPMOVE(<0,PT4,0,0>);
   DPMOVE(<PT1,PT1,0,0>);
   DPMOVE(<PT2,0,0,0>);
   DPMOVE(<PT1,-PT1,0,0>);
   DPMOVE(<0,-PT4,0,0>);
   DPMOVE(<-PT1,-PT1,0,0>);
   ZMOVE(UP);
   DPMOVE(<0,PT1,0,0>);
   ZMOVE(DOWN);
   DPMOVE(<PT1,-PT1,0,0>);
   ZMOVE(UP);
   BRANCH(DONE2);
R:
   ZMOVE(DOWN);
   DPMOVE(<0,PT6,0,0>);
   DPMOVE(<PT3,0,0,0>);
   DPMOVE(<PT1,-PT1,0,0>);
   DPMOVE(<0,-PT1,0,0>);
   DPMOVE(<-PT1,-PT1,0,0>);
   DPMOVE(<-PT3,0,0,0>);
   DPMOVE(<PT4,-PT3,0,0>);
   ZMOVE(UP);
   BRANCH(DONE2);
S:
   DPMOVE(<PT4,PT4,0,0>);
   ZMOVE(DOWN);
   DPMOVE(<0,PT1,0,0>);
   DPMOVE(<-PT1,PT1,0,0>);
   DPMOVE(<-PT2,0,0,0>);
   DPMOVE(<-PT1,-PT1,0,0>);
   DPMOVE(<0,-PT1,0,0>);
```

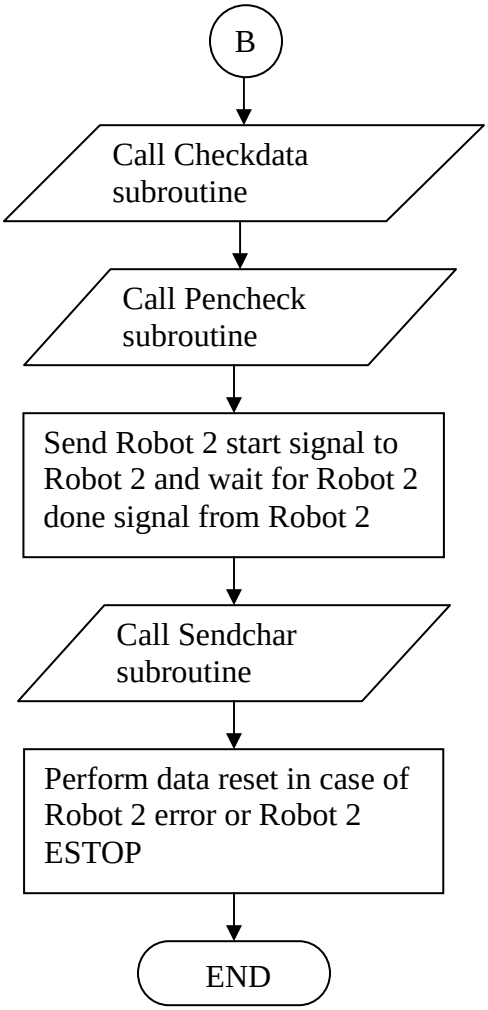
```
    DPMOVE(<PT1,-PT1,0,0>);
    DPMOVE(<PT2,0,0,0>);
    DPMOVE(<PT1,-PT1,0,0>);
    DPMOVE(<0,-PT1,0,0>);
    DPMOVE(<-PT1,-PT1,0,0>);
    DPMOVE(<-PT2,0,0,0>);
    DPMOVE(<-PT1,PT1,0,0>);
    DPMOVE(<0,PT1,0,0>);
    ZMOVE(UP);
    BRANCH(DONE2);
T:
    DPMOVE(<0,PT6,0,0>);
    ZMOVE(DOWN);
    DPMOVE(<PT4,0,0,0>);
    ZMOVE(UP);
    DPMOVE(<-PT2,0,0,0>);
    ZMOVE(DOWN);
    DPMOVE(<0,-PT6,0,0>);
    ZMOVE(UP);
    BRANCH(DONE2);
U:
    DPMOVE(<0,PT6,0,0>);
    ZMOVE(DOWN);
    DPMOVE(<0,-PT5,0,0>);
    DPMOVE(<PT1,-PT1,0,0>);
    DPMOVE(<PT2,0,0,0>);
    DPMOVE(<PT1,PT1,0,0>);
    DPMOVE(<0,PT5,0,0>);
    ZMOVE(UP);
    BRANCH(DONE2);
V:
    DPMOVE(<0,PT6,0,0>);
    ZMOVE(DOWN);
    DPMOVE(<PT2,-PT6,0,0>);
    DPMOVE(<PT2,PT6,0,0>);
    ZMOVE(UP);
    BRANCH(DONE2);
W:
    DPMOVE(<0,PT6,0,0>);
    ZMOVE(DOWN);
    DPMOVE(<PT1,-PT6,0,0>);
    DPMOVE(<PT1,PT3,0,0>);
    DPMOVE(<PT1,-PT3,0,0>);
    DPMOVE(<PT1,PT6,0,0>);
    ZMOVE(UP);
    BRANCH(DONE2);
```

```
X:
   DPMOVE(<0,PT6,0,0>);
   ZMOVE(DOWN);
   DPMOVE(<PT4,-PT6,0,0>);
   ZMOVE(UP);
   DPMOVE(<0,PT6,0,0>);
   ZMOVE(DOWN);
   DPMOVE(<-PT4,-PT6,0,0>);
   ZMOVE(UP);
   BRANCH(DONE2);
Y:
   DPMOVE(<0,PT6,0,0>);
   ZMOVE(DOWN);
   DPMOVE(<PT2,-PT3,0,0>);
   ZMOVE(UP);
   DPMOVE(<-PT2,-PT3,0,0>);
   ZMOVE(DOWN);
   DPMOVE(<PT4,PT6,0,0>);
   ZMOVE(UP);
   BRANCH(DONE2);
Z:
   DPMOVE(<0,PT6,0,0>);
   ZMOVE(DOWN);
   DPMOVE(<PT4,0,0,0>);
   DPMOVE(<-PT4,-PT6,0,0>);
   DPMOVE(<PT4,0,0,0>);
   ZMOVE(UP);
   BRANCH(DONE2);
OVER:
  DELAY(2);
  WRITEO(ROBO2DONE,0);
  END;
```

Appendix 13: Flow chart for Main program




A
Release the pallet from station 0, read the pallet present at station 1 signal and stop and hold the pallet at station 1
Send Robot 1 start signal to Robot 1 and wait for Robot 1 done signal from Robot 1
Send chardone= 1 to PLC and increment the “printed characters” counter
Release the pallet from station 1 when Robot 1 is done and the station 2 does not have any pallet
Read the pallet present at station 2 signal and stop and hold the pallet at station 2
Read the Enter Data signal from HMI
B

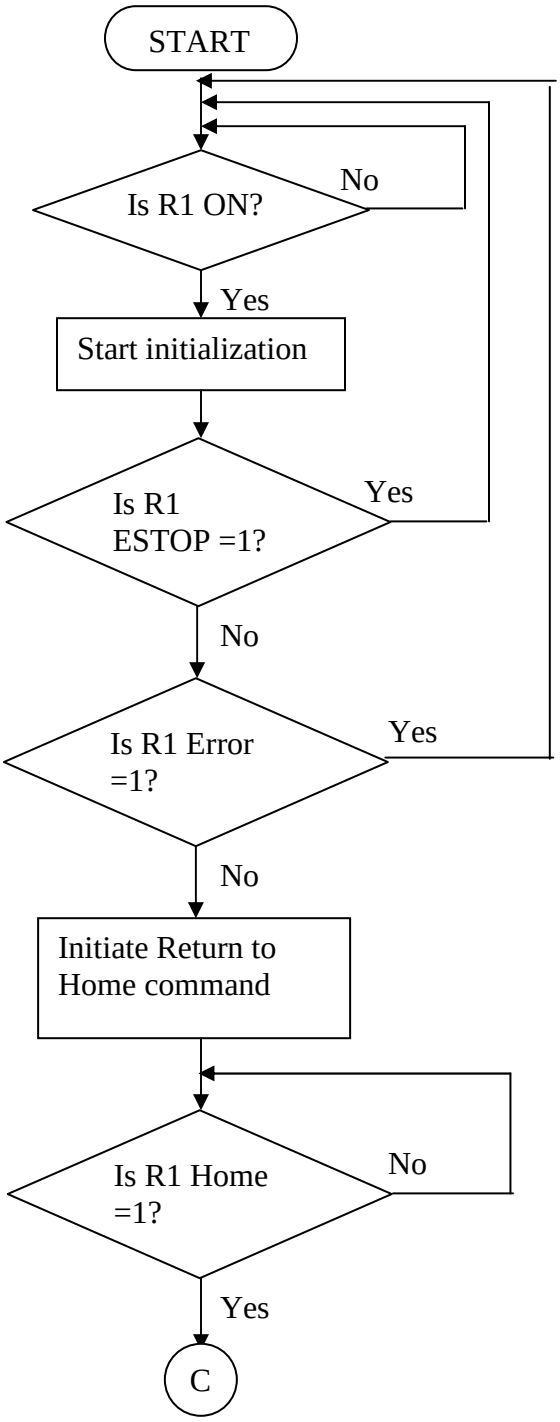

B
Call Checkdata subroutine
Call Pencheck subroutine
Send Robot 2 start signal to Robot 2 and wait for Robot 2 done signal from Robot 2
Call Sendchar subroutine
Perform data reset in case of Robot 2 error or Robot 2 ESTOP
END

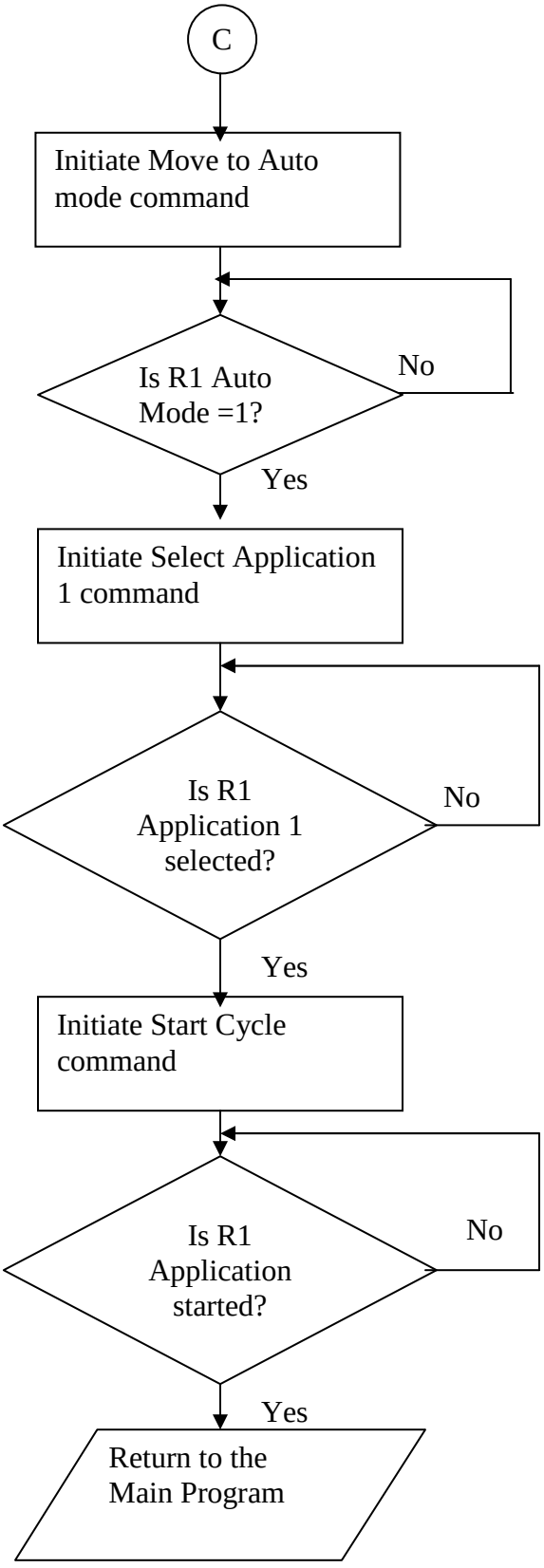
Appendix 14: Flow chart and function details for R1 Startup subroutine

Functions:

- Initialize the Robot 1
- Return to Home position
- Change to Auto mode
- Select the application
- Start the application
- Reset internal Error

Flow Chart


START
Is R1 ON?
No
Yes
Start initialization
Is R1
ESTOP =1?
Yes
No
Is R1 Error
=1?
Yes
No
Initiate Return to
Home command
Is R1 Home
=1?
No
Yes
C

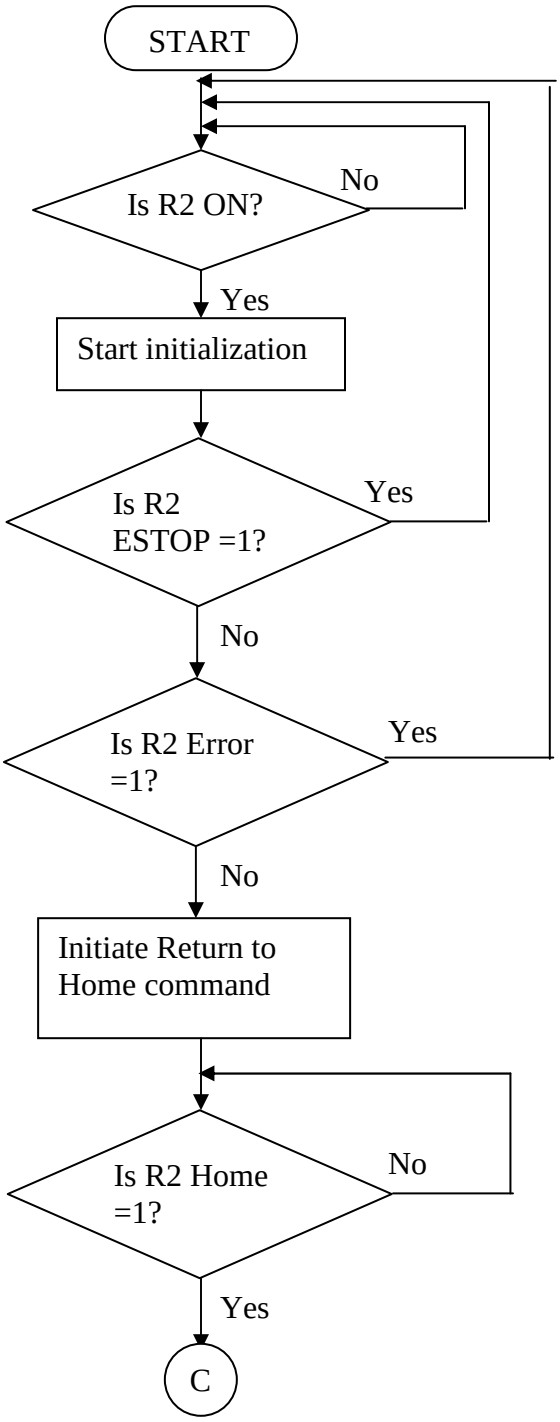

C
Initiate Move to Auto mode command
Is R1 Auto Mode =1?
No
Yes
Initiate Select Application 1 command
Is R1 Application 1 selected?
No
Yes
Initiate Start Cycle command
Is R1 Application started?
No
Yes
Return to the Main Program

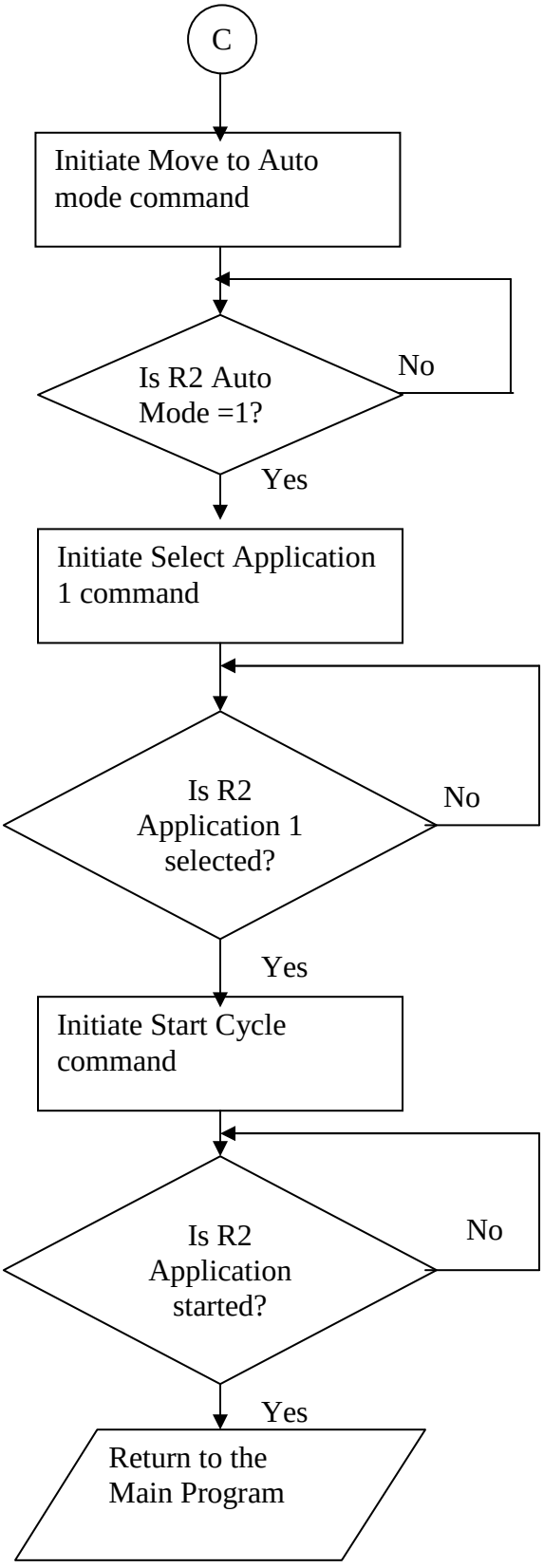
Appendix 15: Flow chart and function details for R2 Startup subroutine

Functions:

- Initialize the Robot 2
- Return to Home position
- Change to Auto mode
- Select the application
- Start the application
- Reset internal Error

Flow Chart:


START
Is R2 ON?
No
Yes
Start initialization
Is R2
ESTOP =1?
Yes
No
Is R2 Error
=1?
Yes
No
Initiate Return to
Home command
Is R2 Home
=1?
No
Yes
C

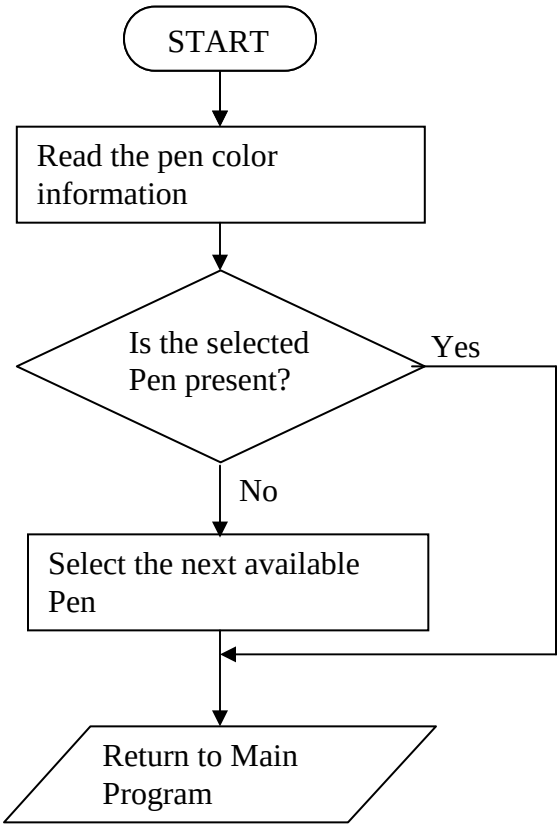

C
Initiate Move to Auto mode command
Is R2 Auto Mode =1?
No
Yes
Initiate Select Application 1 command
Is R2 Application 1 selected?
No
Yes
Initiate Start Cycle command
Is R2 Application started?
No
Yes
Return to the Main Program

Appendix 16: Flow chart and function details for Pen Check subroutine

Functions:

- Reads the selected pen data from HMI screen
- Checks if the selected pen is available on the Pen feeder station
- If not, generate the pen absent signal
- In case the pen is not available, sends the default pen value to the file

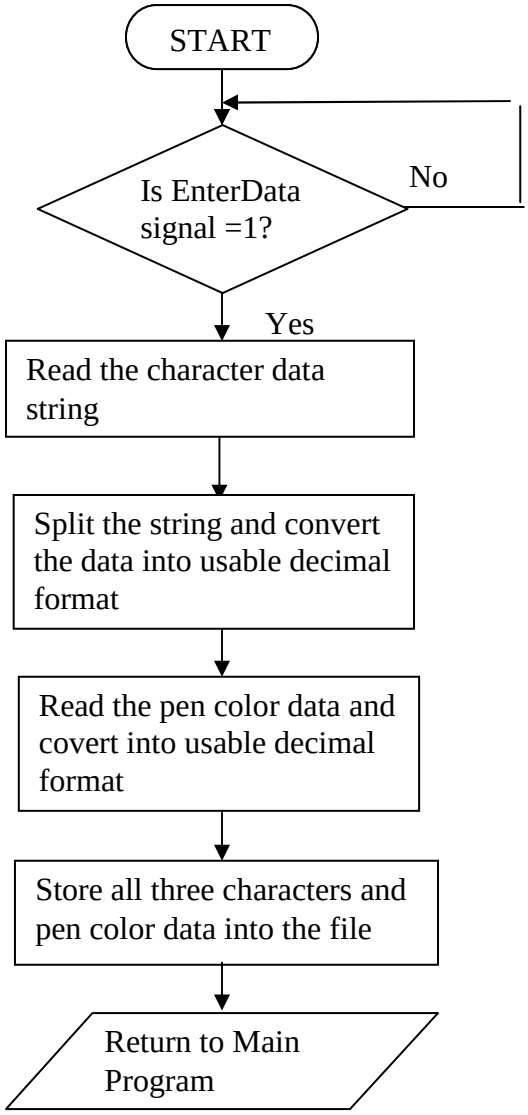
Flow Chart



Appendix 17: Flow chart and function details for Pen Check subroutine

Functions:

- Reads the characters entered from the HMI screen
- Validates the entered value
- Converts this information into the decimal number
- Loads the validated values into the file

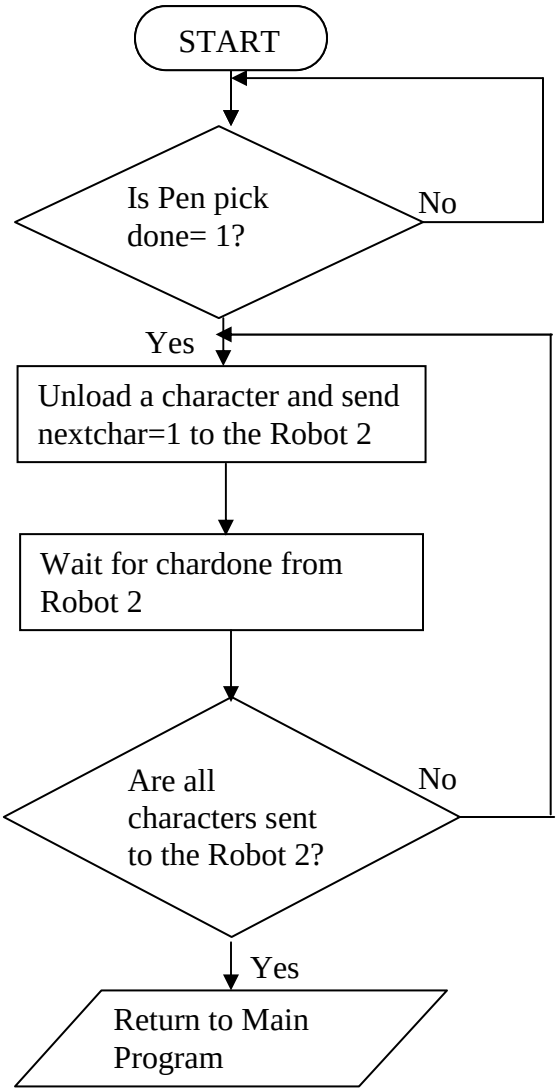


Appendix 18: Flow chart and function details for Send Char subroutine

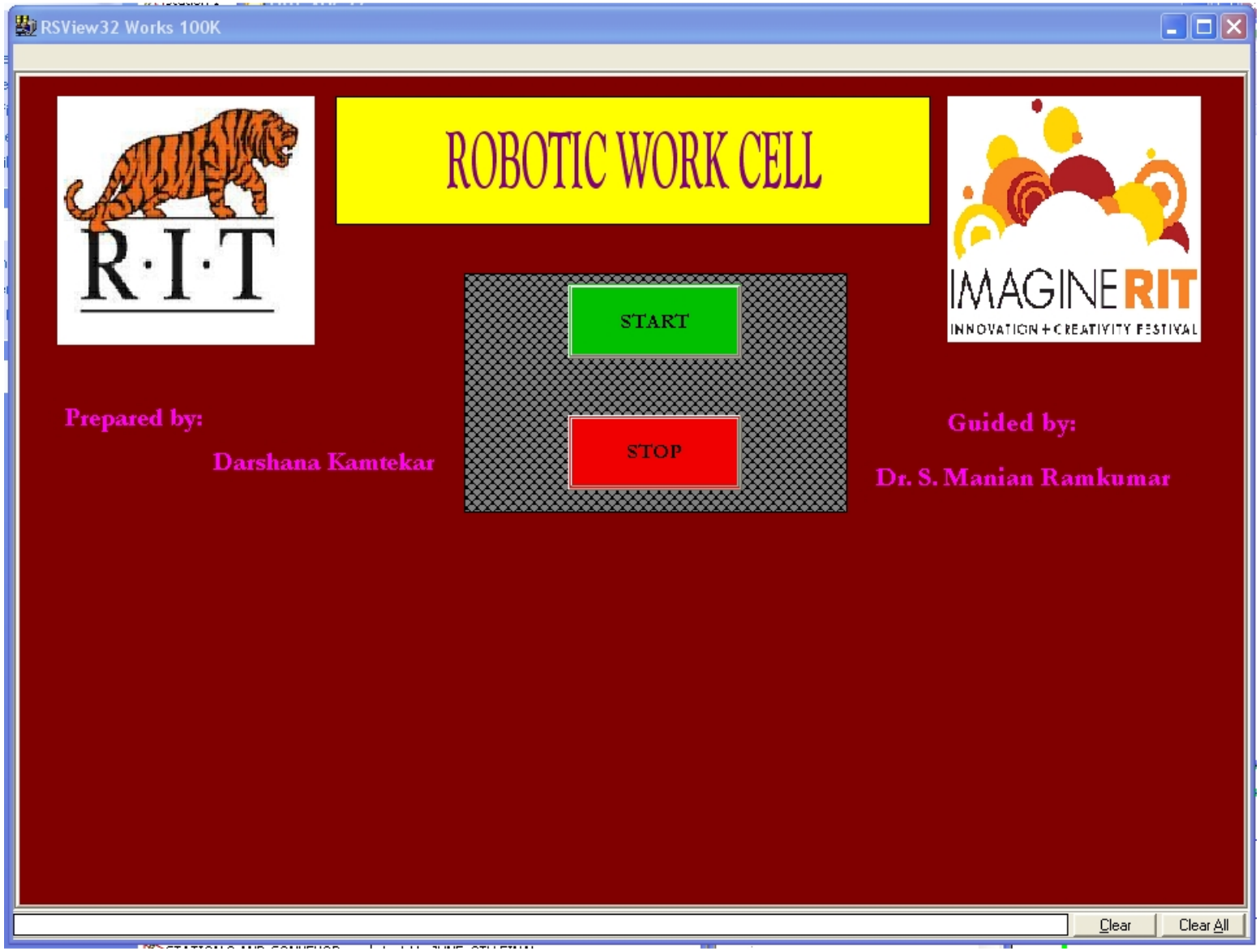
Functions:

- Unloads all three characters subsequently
- Sends the equivalent binary count to the Robot 2
- Synchronizes the charactering sending with Robot 2 chardone and nextchar signals

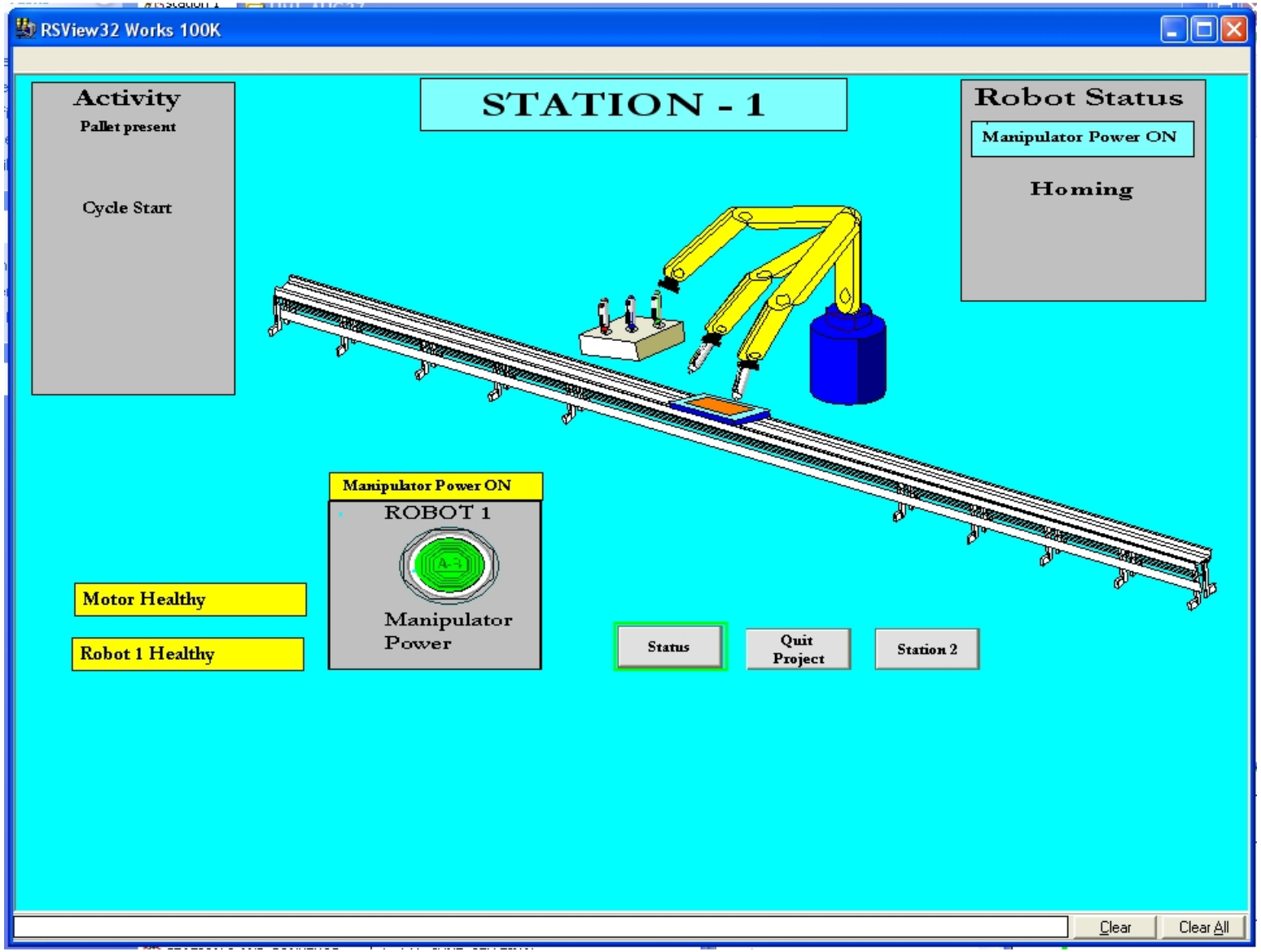
Flow Chart



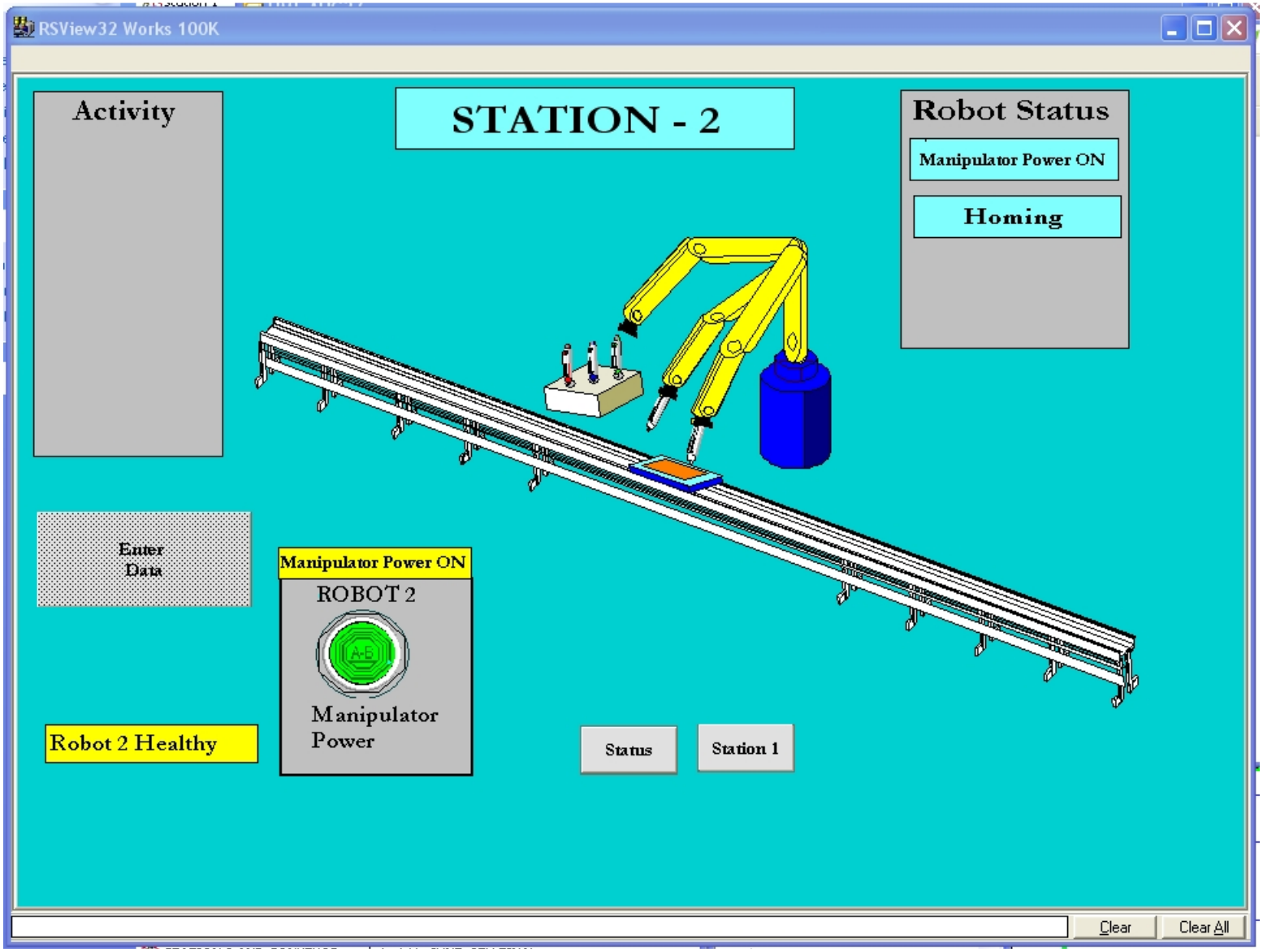
Appendix 19: HMI Display Screen- Main



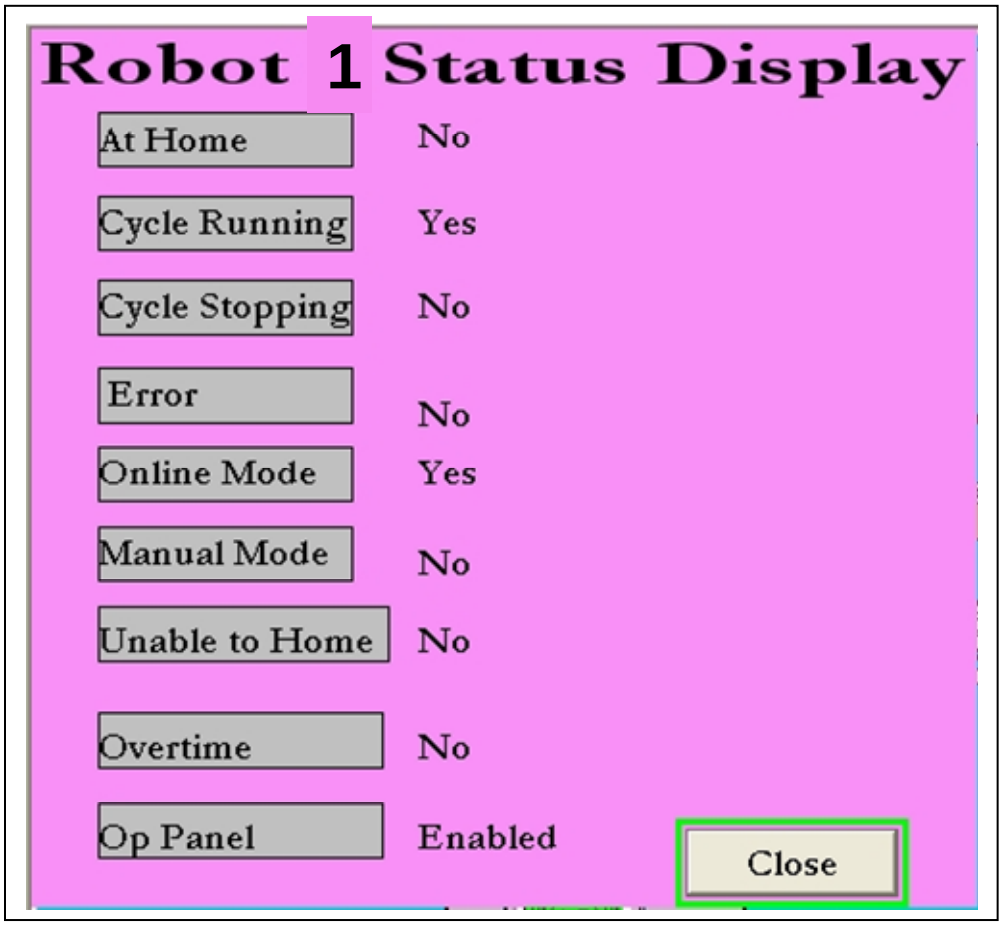
Appendix 20: HMI Display Screen- Station 1



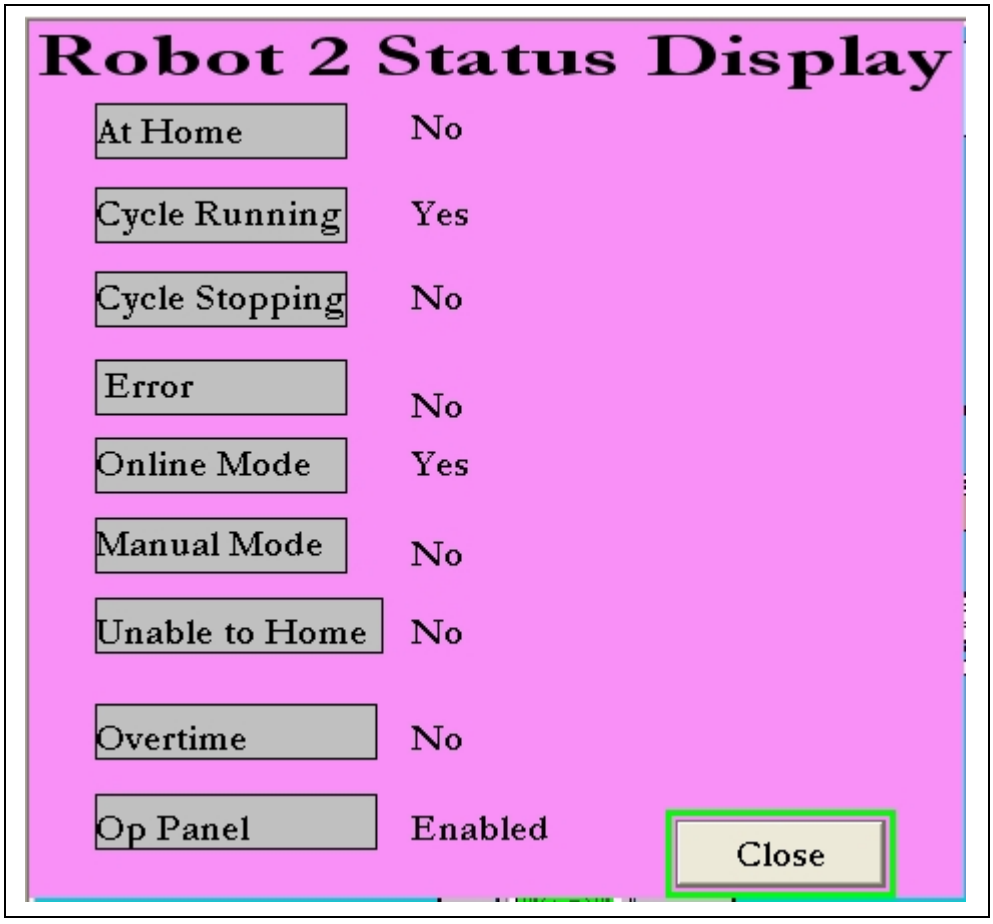
Appendix 21: HMI Display Screen- Station 2



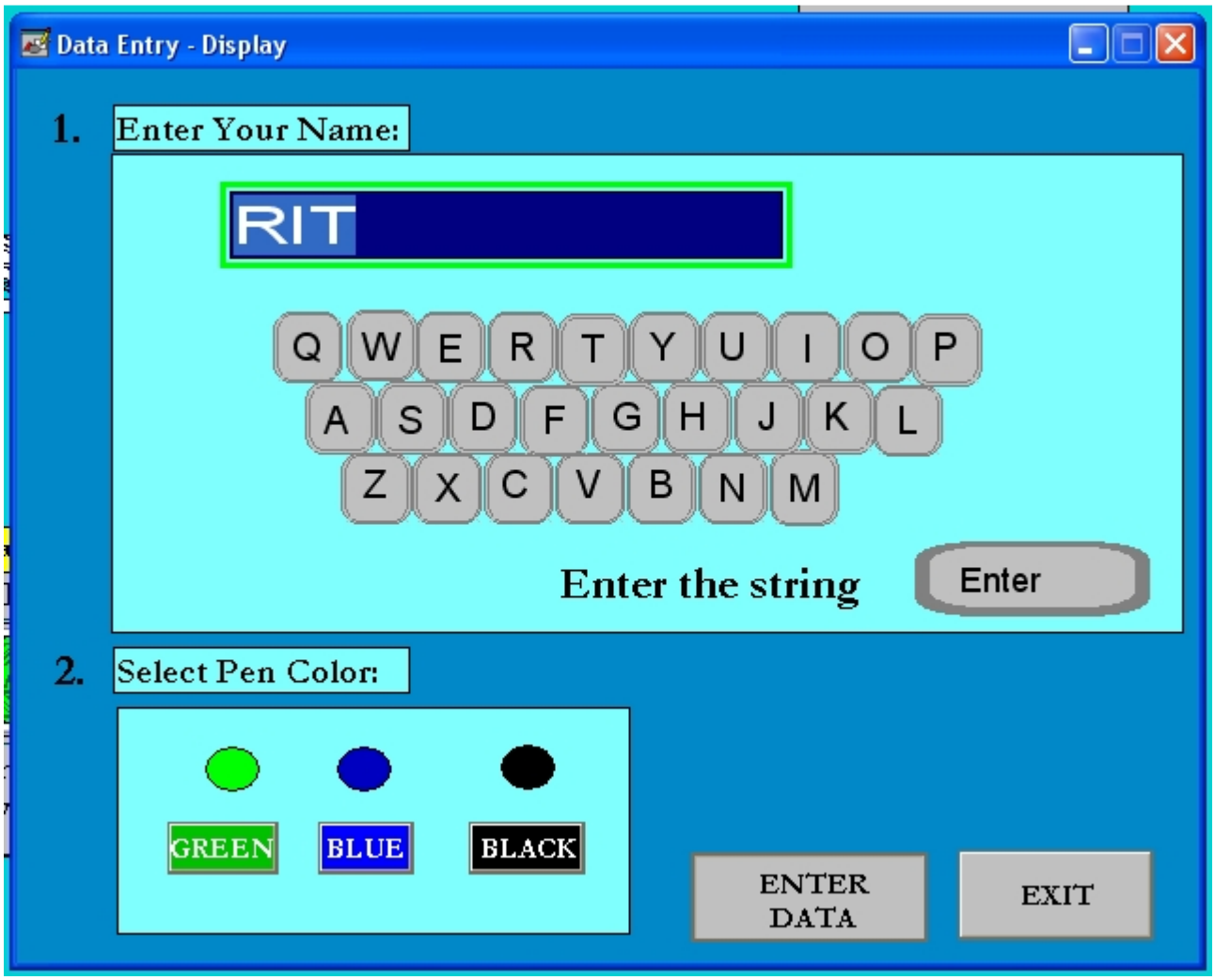
Appendix 22: HMI Display Screen- Station 1 Status



Appendix 23: HMI Display Screen- Station 2 Status



Appendix 24: HMI Display Screen- Data Entry



Appendix 25: List of Siemens Safety I/O modules and Siemens safety sensors

Safety I/Os:

Safety system Rack	**Module No.**	**Module Name**	**Module Type**	**Part No.**
Safety CPU Rack	PS307-5A	Power Supply, 5A	General Purpose	6ES7 307-1EA00-0AA0
	CPU 315F-2PN/DP	CPU	Safety	6ES7 315-2FH13-0AB0
	Safety Protector	Safety Protector	Safety	195-7KF00-0XA0
	SM 326 DI 24 X DC24V	24 ch Digital Input Module	Safety	6ES7 326-1BK01-0AB0
	SM 326, 8 DO; DC 24V	8 ch Digital Output Module	Safety	6ES7 326-2BF40-0AB0
Distributed Safety I/O Rack	IM 151- 3 PN	Distributed I/O Head Module	General Purpose	6ES7 151-3BA22-0AB0
	PM-E 24V	Power Terminal Module	General Purpose	6ES7 138-4CA50-0AB0
	2DI-HF- 1	2 ch Digital Input Module	General Purpose	6ES7 131-4BB01-0AB0
	2DI-HF- 2	2 ch Digital Input Module	General Purpose	6ES7 131-4BB01-0AB0
	2DI-HF- 3	2 ch Digital Input Module	General Purpose	6ES7 131-4BB01-0AB0
	PM-E 24V	Power Terminal Module	General Purpose	6ES7 138-4CA50-0AB0
	4/8 F-DI- 1	4/ 8 ch Digital Input Module	Safety	6ES7 138-4FA03-0AB0
	4/8 F-DI- 2	4/ 8 ch Digital Input Module	Safety	6ES7 138-4FA03-0AB0
	PM-E 24V	Power Terminal Module	General Purpose	6ES7 138-4CA50-0AB0
	4 F-DO	4 ch Digital Output Module	Safety	6ES7 138-4FB02-0AB0

	PM-E 24V	Power Terminal Module	General Purpose	6ES7 138-4CA50-0AB0
	2 DO-HF- 1	2 ch Digital Output Module	General Purpose	6ES7 132-4BB01-0AB0
	2 DO-HF- 2	2 ch Digital Output Module	General Purpose	6ES7 132-4BB01-0AB0
	2 DO-HF- 3	2 ch Digital Output Module	General Purpose	6ES7 132-4BB01-0AB0
	Terminator	Terminator Module	General Purpose	

Safety Sensors:

Test case 1:

Sensor Name	Description	Sensor Type	Part No.
K11	Contactor with Permanent Aux Contacts	Safety	3RT1015-1BB44-3MA0
K12	Contactor with Permanent Aux Contacts	Safety	3RT1015-1BB44-3MA0
DG	5-way actuation Position Switch with 2 NC contacts	Safety	3SE3 200-6XX13

Test case 2A+2B:

Sensor Name	Description	Sensor Type	Part No.
K21	Contactor with Permanent Aux Contacts	Safety	3RT1015-1BB44-3MA0
K22	Contactor with Permanent Aux Contacts	Safety	3RT1015-1BB44-3MA0
LCT	Transmitter for Light Curtains	Safety	3RG78454DB00
LCR	Receiver for Light Curtains	Safety	3RG78454DB01

Test case 3:

Sensor Name	Description	Sensor Type	Part No.
K31	Contactor with Permanent Aux Contacts	Safety	3RT1015-1BB44-3MA0
K32	Contactor with Permanent Aux Contacts	Safety	3RT1015-1BB44-3MA0
Mag. Door 1	Magnetically operated switch	Safety	3SE6 704-2BA
		Safety	3SE6 604-2BA
Mag. Door 2	Magnetically operated switch	Safety	3SE6 704-3BA
		Safety	3SE6 605-3BA

Common:

Sensor Name	Description	Sensor Type	Part No.
START	Green ILPB with NO Contact	General Purpose	3SB30010AA41
STOP	Red ILPB with NC Contact	General Purpose	3SB30010AA21
ACK	Blue ILPB with NO Contact	General Purpose	3SB30010AA51
Muting Lamp	Indicator Light with holder, Clear	General Purpose	3SB30016AA70
MS_11	Toggle switch	General Purpose	
MS_12	Toggle switch	General Purpose	
MS_21	Toggle switch	General Purpose	
MS_22	Toggle switch	General Purpose	
ESTOP 12	SIGNUM 3SB3 ACTUATOR EMERGENCY-STOP with 2 NC contacts	Safety	3SB35001FA20
ESTOP 3	E-Stop Pushbutton Station with 2 NC contacts	Safety	3SB38010DG3
ESTOP ALL	E-Stop Pushbutton Station with 2 NC contacts	Safety	3SB38010DG3

Appendix 26: The I/O list for safety PLC and associated sensors connections

Signal Name	Module No.	IO Type	Channel No.	Voting Configuration	Address
START	SM 326 DI 24 X DC24V	DI	0	1oo1	I 0.0
ACK	SM 326 DI 24 X DC24V	DI	1	1oo1	I 0.1
STOP	SM 326 DI 24 X DC24V	DI	2	1oo1	I 0.2
MS_11	SM 326 DI 24 X DC24V	DI	12	1oo1	I 1.4
MS_12	SM 326 DI 24 X DC24V	DI	13	1oo1	I 1.5
MS_21	SM 326 DI 24 X DC24V	DI	14	1oo1	I 1.6
MS_22	SM 326 DI 24 X DC24V	DI	15	1oo1	I 1.7
ESTOP-ALL	SM 326 DI 24 X DC24V	DI	6, 18	1oo2	I 0.4
OSSD	SM 326 DI 24 X DC24V	DI	7, 19	1oo2	I 0.7
Mag. Door	SM 326 DI 24 X DC24V	DI	8, 20	1oo2	I 1.0
Muting Lamp	SM 326, 8 DO; DC 24V	DO	0	Readback	Q 10.0
START Lamp	SM 326, 8 DO; DC 24V	DO	1	Readback	Q 10.1
ACK Lamp	SM 326, 8 DO; DC 24V	DO	4	Readback	Q 10.4
STOP Lamp	SM 326, 8 DO; DC 24V	DO	5	Readback	Q 10.5
K11_NC	2DI-HF- 1	DI	0	1oo1	I 42.0
K12_NC	2DI-HF- 1	DI	1	1oo1	I 42.1
K21_NC	2DI-HF- 2	DI	0	1oo1	I 43.0
K22_NC	2DI-HF- 2	DI	1	1oo1	I 43.1
K31_NC	2DI-HF- 3	DI	0	1oo1	I 37.0
K32_NC	2DI-HF- 3	DI	1	1oo1	I 37.1

Appendix 27: The modified I/O list after implementation of the "After Case" system

Rack #2 Connections:

RACK #2 (ROBOT 1) DC Module				
Datafile (Octal)	**PLC Rack**	**Description**	**Type**	**External Device**
I:020/0	I:020/0	R1 Cycle Done	Robot Output	Robot DO 001 (q)
I:020/1	I:020/1	R1 Inspection Done	Robot Output	Robot DO 002 (r)
I:020/2	I:020/2	End Effector Error Robot1: Send	Robot Output	Robot DO 003 (s)
I:020/3	I:020/3	R1_Healthy		
I:020/4	I:020/4	Motor_Healthy		
I:020/5	I:020/5			
I:020/6	I:020/6	Cycle Running- Robot 1	Robot Internal Status Output bit	Robot DO (e)
I:020/7	I:020/7	Error - Robot 1		Robot DO (f)
I:020/10	I:020/8	At Home - Robot 1		Robot DO (g)
I:020/11	I:020/9	Unable to move Home - Robot 1	Robot Internal Output	Robot DO (h)
I:020/12	I:020/10	Manipulator Power ON - Robot 1		Robot DO11 (AA)
I:020/13	I:020/11	Online - Robot 1		Robot DO12 (AB)
I:020/14	I:020/12	Manual Mode - Robot 1		Robot DO13 (AC)
I:020/15	I:020/13	Cycle stopping - Robot 1		Robot DO14 (AD)
I:020/16	I:020/14	Overtime - Robot 1		Robot DO15 (AE)
I:020/17	I:020/15	Op Panel Disabled - Robot 1		Robot DO16 (AF)

Datafile (Octal)	PLC Rack	Description	Type	External Device
O:020/0	O:020/0	R1 Error Clear	Robot Input	Robot DI 001(A)
O:020/1	O:020/1	R1 Start Cycle	Robot Input	Robot DI 002(B)
O:020/2	O:020/2			
O:020/3	O:020/3			
O:020/4	O:020/4			
O:020/5	O:020/5			
O:020/6	O:020/6			
O:020/7	O:020/7	Inhibit Move to Home - Robot 1	Robot Internal Input	Robot DI (W) (Grnd)
O:020/10	O:020/8			Robot DI (X) (Grnd)
O:020/11	O:020/9	Manipulator Power(Y&Z)- Robot 1		Robot DI (Y,Z)
O:020/12	O:020/10	Command Strobe - Robot 1	Robot Internal Command Input bits	Robot DI (a)
O:020/13	O:020/11	Command Bit 0 - Robot 1		Robot DI 012 (M)
O:020/14	O:020/12	Command Bit 1 - Robot 1		Robot DI 013 (N)
O:020/15	O:020/13	Command Bit 2 - Robot 1		Robot DI 014 (O)
O:020/16	O:020/14	Command Bit 3 - Robot 1		Robot DI 015 (P)
O:020/17	O:020/15	Command Bit 4 - Robot 1		Robot DI 016 (R)
End Effector Robot1			Robot D1 003 (C)	
Input Ground Robot 1			Robot DI (S, T, U, V)	

Output Ground Robot 1	Robot DO (j, k, l, m, n, p)
Controller Frame Ground Robot 1	Robot (AG)

Rack #4 Connections:

RACK #4 (ROBOT 2) DC Module				
Datafile (Octal)	**PLC Rack**	**Description**	**Type**	**External Device**
I:040/0	I:040/0	R2 Cycle Done	Robot Output	Robot DO 001 (q)
I:040/1	I:040/1	Pen picked (Done)	Robot Output	Robot DO 002 (r)
I:040/2	I:040/2	Char Done	Robot Output	Robot DO 003 (s)
I:040/3	I:040/3	End Effector Error Robot2: Send	Robot Output	Robot DO 004 (t)
I:040/4	I:040/4	R2_Healthy		
I:040/5	I:040/5			
I:040/6	I:040/6	Cycle Running - Robot 2	Robot Internal Status Output bit	Robot DO (e)
I:040/7	I:040/7	Error- Robot 2		Robot DO (f)
I:040/10	I:040/8	At Home- Robot 2		Robot DO (g)
I:040/11	I:040/9	Unable to move Home -Robot 2	Robot Internal Output	Robot DO (h)
I:040/12	I:040/10	Manipulator power ON- Robot 2		Robot DO 011 (AA)
I:040/13	I:040/11	Online- Robot 2		Robot DO 012 (AB)
I:040/14	I:040/12	Manual Mode- Robot 2		Robot DO 013 (AC)
I:040/15	I:040/13	Cycle stopping- Robot 2		Robot DO 014 (AD)
I:040/16	I:040/14	Overtime- Robot 2		Robot DO 015 (AE)
I:040/17	I:040/15	Op Panel Disabled- Robot 2		Robot DO 016 (AF)

Datafile (Octal)	PLC Rack	Description	Type	External Device
O:040/0	O:040/0	Char bit 0/ Pen select 1	Robot Input	Robot DI 001 (A)
O:040/1	O:040/1	Char bit 1/Pen select 2	Robot Input	Robot DI 002 (B)
O:040/2	O:040/2	Char bit 2	Robot Input	Robot DI 003 (C)
O:040/3	O:040/3	Char bit 3	Robot Input	Robot DI 004 (D)
O:040/4	O:040/4	Char bit 4	Robot Input	Robot DI 005 (E)
O:040/5	O:040/5	Read Next Char	Robot Input	Robot DI 006 (F)
O:040/6	O:040/6	R2 Error Clear: end effector	Robot Input	Robot DI 009 (I)
O:040/7	O:040/7	R2 Start Cycle/Pen check done	Robot Input	Robot DI 007 (G)
O:040/10	O:040/8		Robot Internal Input	Robot DI (X) (Grnd)
O:040/11	O:040/9	Manipulator Power(Y&Z) - Robot 2		Robot DI (Y,Z)
O:040/12	O:040/10	Command Strobe - Robot 2	Robot Internal Command Input bits	Robot DI (a)
O:040/13	O:040/11	Command Bit 0 - Robot 2		Robot DI 012 (M)
O:040/14	O:040/12	Command Bit 1 - Robot 2		Robot DI 013 (N)
O:040/15	O:040/13	Command Bit 2 - Robot 2		Robot DI 014 (O)
O:040/16	O:040/14	Command Bit 3 - Robot 2		Robot DI 015 (P)
O:040/17	O:040/15	Command Bit 4 - Robot 2		Robot DI 016 (R)
End Effector Robot 2			Robot DI 008 (H)	
Input Ground Robot 2			Robot DI (S, T, U, V)	
Output Ground Robot 2			Robot DO (j, k, m, n, p)	
Controller Frame Ground Robot 2			Robot (AG)	

Rack #1 Connections:

RACK #1 (ROBOT 3) DC Module				
Datafile (Octal)	**PLC Rack**	**Description**	**Type**	**External Device**
I:010/0	I:010/0	R3 Cycle Done	Robot Output	Robot DO 001 (q)
I:010/1	I:010/1	**********Not used**********		
I:010/2	I:010/2	End Effector Error Robot 3	Robot Output	Robot DO 002 (r)
I:010/3	I:010/3			
I:010/4	I:010/4			
I:010/5	I:010/5			
I:010/6	I:010/6	Cycle running- Robot 3	Robot Internal Status Output bit	Robot DO (e)
I:010/7	I:010/7	Error- Robot 3		Robot DO (f)
I:010/10	I:010/8	At home- Robot 3		Robot DO (g)
I:010/11	I:010/9	Unable to move Home -Robot 3	Robot Internal Output	Robot DO (h)
I:010/12	I:010/10	Manipulator power ON- Robot 3		Robot DO 011 (AA)
I:010/13	I:010/11	Online- Robot 3		Robot DO 012 (AB)
I:010/14	I:010/12	Manual Mode- Robot 3		Robot DO 013 (AC)
I:010/15	I:010/13	Cycle stopping- Robot 3		Robot DO 014 (AD)
I:010/16	I:010/14	Overtime- Robot 3		Robot DO 015 (AE)

I:060/17	I:010/15	Op Panel Disabled- Robot 3		Robot DO 016 (AF)

Datafile (Octal)	**PLC Rack**	**Description**	**Type**	**External Device**
O:010/0	O:010/0	Start Cycle to Robot 3	Robot Input	Robot DI 001 (A)
O:010/1	O:010/1	Error Clear : Robot 3	Robot Input	Robot DI 002 (B)
O:010/2	O:010/2	Bad from Vision system (1/0)	Camera output	Robot DI 003 (C)
O:010/3	O:010/3	Good from Vision system (1/0)	Camera output	Robot DI 003 (D)
O:010/4	O:010/4			
O:010/5	O:010/5			
O:010/6	O:010/6	Inhibit Move to Home - Robot 2	Robot Internal Input	Robot DI (W) (Grnd)
O:010/7	O:010/7	Inhibit Move to Home - Robot 3		Robot DI (W) (Grnd)
O:010/10	O:010/8	Emergency Stop - Robot 3		Robot DI (X) (Grnd)
O:010/11	O:010/9	Manipulator Power(Y&Z) - Robot 3		Robot DI (Y,Z)
O:010/12	O:010/10	Command Strobe - Robot 3	Robot Internal Command Input bits	Robot DI (a)
O:010/13	O:010/11	Command Bit 0 - Robot 3		Robot DI 012 (M)
O:010/14	O:010/12	Command Bit 1 - Robot 3		Robot DI 013 (N)
O:010/15	O:010/13	Command Bit 2 - Robot 3		Robot DI 014 (O)
O:010/16	O:010/14	Command Bit 3 - Robot 3		Robot DI 015 (P)
O:060/17	O:010/15	Command Bit 4 - Robot 3		Robot DI 016 (R)

End Effector Robot 3	Robot D1 004 (D)
Input Ground Robot 3	Robot DI (S, T, U, V)
Output Ground Robot 3	Robot DO (j, k, m, n, p)
Controller Frame Ground Robot 3	Robot (AG)

Rack #6 Connections:

RACK #6 (Conveyor) DC Module				
Datafile (Octal)	**PLC Rack**	**Description**	**Type**	**External Device**
I:060/0	I:060/0	Pallet Present @ Station 0	On Conveyor	Proximity sensor
I:060/1	I:060/1	Pallet Present @ Station 1		
I:060/2	I:060/2	Pallet Present @ Station 2		
I:060/3	I:060/3	Pallet Present @ Station 3		
I:060/4	I:060/4	Pallet Present @ Station 4		
I:060/5	I:060/5	Pallet placed correctly		
I:060/6	I:060/6	Pallet present before st0		Light beam sensor
I:060/7	I:060/7	Paper Stack Empty	On Paper Feeder	
I:060/10	I:060/8	Pen present 1 @ feeder 2	On Pen Feeder	
I:060/11	I:060/9	Pen present 2 @ feeder 2		
I:060/12	I:060/10	Pen present 3 @ feeder 2		

I:060/13	I:060/11	output 1: pass		Vision System
I:060/14	I:060/12	output 2: busy		
I:060/15	I:060/13			
I:060/16	I:060/14			
I:060/17	I:060/15			

Datafile (Octal)	**PLC Rack**	**Description**	**Type**	**External Device**
O:060/0	O:060/0	Motor Running	Relay Logic	motor
O:060/1	O:060/1	Trigger		
O:060/2	O:060/2			
O:060/3	O:060/3			
O:060/4	O:060/4			
O:060/5	O:060/5			
O:060/6	O:060/6			
O:060/7	O:060/7			
O:060/10	O:060/8			
O:060/11	O:060/9			
O:060/12	O:060/10			

O:060/13	O:060/11			
O:060/14	O:060/12			
O:060/15	O:060/13			
O:060/16	O:060/14			
O:060/17	O:060/15			

Rack #3 Connections:

RACK #3 (Conveyor) AC Module				
Datafile	**PLC Rack**	**Description**	**Type**	**External Device**
I:030/0	SAME as OCTAL	Motor Input	relay logic	From the start and stop switches
I:030/1				
I:030/2				
I:030/3				
I:030/4				
I:030/5				
I:030/6				
I:030/7				
I:030/10				
I:030/11				

I:030/12				
I:030/13				
I:030/14				
I:030/15				
I:030/16				
I:030/17				

Datafile	**PLC Rack**	**Description**	**Type**	**External Device**
O:030/0	SAME as OCTAL	Stopper at Station0	Pneumatics	On Conveyor
O:030/1		Stopper at Station1		
O:030/2		Stopper at Station2		
O:030/3		Stopper at Station3		
O:030/4		Stopper at Station4		
O:030/5		Clamp at Station1		
O:030/6		Clamp at Station2		
O:030/7		Pneumatic Clamp at Station3		
O:030/10				
O:030/11				
O:030/12				
O:030/13				
O:030/14				

O:030/15				
O:030/16				
O:030/17				

Appendix 28: Safety PLC Program

Main Project 1\ SIMATIC 300 (1)\ CPU 315F- 2 PN /DP\ S7 Program (1)

Safety program 'S7 Program(1)' - Offline

Collective signature

F blocks with F attribute of the block container:	23647BE1
Safety program:	23647BE1

Distributed Safety Version

Version ID:	V5.4+SP3

Current generation

Generation time:	10/01/2009 11:58:52 AM

Blocks in safety program

F block	Symbolic name	Function in safety program	Signature	Initial value signature
FC1	F_CALL	F-CALL	929A	
FC2	F_Reintegration	F-FC	D30E	
FC10	Safety_Prgm	F-program block	CC2C	
FB1	F_FEEDBACK	F-FB	8401	5904
FB2	F_Light Curtain	F-FB	C371	5904
FB3	F_Motor Control	F-FB	3D83	5904
FB186	F_TOF	F application block	14B4	980D
FB189	F_MUTING	F application block	606B	AF14
FB216	F_FDBACK	F application block	F521	F965
FB1638	F_IO_CGP	F-system block	EDA2	DC2F
FB1639	F_CTRL_1	F-system block	504C	BED9
FB1640	F_CTRL_2	F-system block	40BA	9E40
FB1641	FITOF	F-system block	69AF	3326
FB1642	FIMUTING	F-system block	D5F9	B0C4
FB1643	F_DIAG_N	F-system block	99CA	3612
FB1644		Automatically generated F block	AD73	
FB1645		Automatically generated F block	AB34	
FB1646		Automatically generated F block	B541	
FB1647		Automatically generated F block	70F8	
FB1648		Automatically generated F block	A5E3	
FB1649		Automatically generated F block	B0AD	
DB1	Instanz_FB1	I-DB for F-FB	5904	
DB2	Instanz_FB2	I-DB for F-FB	5904	
DB3	Instanz_FB3	I-DB for F-FB	5904	

DB189	Instanz_FB189	I-DB for F-application block	DDA4	
DB216	Instanz_FB216R1	I-DB for F-application block	B91F	
DB217	Instanz_FB216R2	I-DB for F-application block	B91F	
DB218	Instanz_FB216Motor	I-DB for F-application block	B91F	
DB818	F_GLOBDB	F shared DB	BC6E	
DB819	F00000_DI24xDC24V	F I/O DB	C037	
DB820	F00010_DO8xDC24V_2A	F I/O DB	F89D	
DB821	F00015_4_8_F_DI_DC24V	F I/O DB	DB75	
DB822	F00021_4_8_F_DI_DC24V	F I/O DB	C573	
DB823	F00027_4_F_DO_DC24V_2A	F I/O DB	8292	
DB824		Automatically generated F block	9D3D	
DB825		Automatically generated F block	2685	
DB826		Automatically generated F block	87BC	
DB827		Automatically generated F block	4D5E	
DB828		Automatically generated F block	61CC	
DB829		Automatically generated F block	3108	
DB830		Automatically generated F block	61CC	
DB831		Automatically generated F block	61CC	
DB832		Automatically generated F block	9720	
DB833		Automatically generated F block	13E7	
DB834		Automatically generated F block	9720	
DB835		Automatically generated F block	4D5E	
DB836		Automatically generated F block	562A	
DB837		Automatically generated F block	BDB5	

[...] = Block without F attribute

Data from the standard user program

Address	Symbol	F-runtime group
M 90.0	FEEDBACK_R1	FC1
M 90.1	FEEDBACK_R2	FC1
M 90.2	FEEDBACK_MOTOR	FC1
M 90.5	COND_R1	FC1
M 91.0	COND_R2	FC1
M 91.3	COND_MOTOR	FC1
M 92.1	ACK_REQ1	FC1

Runtime group information

F-runtime group FC1

Number of the F-CALL:	FC1
Symbolic name:	F_CALL
Number of the F program block called:	FC10
Symbolic name:	Safety_Prgm

Number of the corresponding instance DB:
Symbolic name:
Maximum cycle time: T#200MS

F-runtime group blocks:

F block	Symbolic name	Function in safety program	Signature	Initial value signature
FC1	F_CALL	F-CALL	929A	
FC2	F_Reintegration	F-FC	D30E	
FC10	Safety_Prgm	F-program block	CC2C	
FB1	F_FEEDBACK	F-FB	8401	5904
FB2	F_Light Curtain	F-FB	C371	5904
FB3	F_Motor Control	F-FB	3D83	5904
FB186	F_TOF	F application block	14B4	980D
FB189	F_MUTING	F application block	606B	AF14
FB216	F_FDBACK	F application block	F521	F965
DB1	Instanz_FB1	I-DB for F-FB	5904	
DB2	Instanz_FB2	I-DB for F-FB	5904	
DB3	Instanz_FB3	I-DB for F-FB	5904	
DB189	Instanz_FB189	I-DB for F-application block	DDA4	
DB216	Instanz_FB216R1	I-DB for F-application block	B91F	
DB217	Instanz_FB216R2	I-DB for F-application block	B91F	
DB218	Instanz_FB216Motor	I-DB for F-application block	B91F	
DB819	F00000_DI24xDC24V	F I/O DB	C037	
DB820	F00010_DO8xDC24V_2A	F I/O DB	F89D	
DB821	F00015_4_8_F_DI_DC24V	F I/O DB	DB75	
DB822	F00021_4_8_F_DI_DC24V	F I/O DB	C573	
DB823	F00027_4_F_DO_DC24V_2A	F I/O DB	8292	

[...] = Block without F attribute

Addressed F I/O:

Symbolic name of the F I/O DB:	F00000_DI24xDC24V
F I/O DB number:	DB819
Initial Address:	0
Name/Description:	DI24xDC24V
Module type:	Input
F_Monitoring time:	200
CRC parameter:	3BD0
F_Source_Address:	1
F_Target_Address:	819
PROFIsafe:	V1 -MODE
Channel-granular passivation:	No
Symbolic name of the F I/O DB:	F00010_DO8xDC24V_2A
F I/O DB number:	DB820
Initial Address:	10
Name/Description:	DO8xDC24V/2A
Module type:	Output
F_Monitoring time:	200

CRC parameter:	1B7A
F_Source_Address:	1
F_Target_Address:	204
PROFIsafe:	V1 -MODE
Channel-granular passivation:	No
Symbolic name of the F I/O DB:	F00015_4_8_F_DI_DC24V
F I/O DB number:	DB821
Initial Address:	15
Name/Description:	4/8 F-DI DC24V
Module type:	Input
F_Monitoring time:	120
CRC parameter:	49A2
F_Source_Address:	2000
F_Target_Address:	199
PROFIsafe:	V2-MODE
Channel-granular passivation:	No
Symbolic name of the F I/O DB:	F00021_4_8_F_DI_DC24V
F I/O DB number:	DB822
Initial Address:	21
Name/Description:	4/8 F-DI DC24V
Module type:	Input
F_Monitoring time:	100
CRC parameter:	35EF
F_Source_Address:	2000
F_Target_Address:	1022
PROFIsafe:	V2-MODE
Channel-granular passivation:	No
Symbolic name of the F I/O DB:	F00027_4_F_DO_DC24V_2A
F I/O DB number:	DB823
Initial Address:	27
Name/Description:	4 F-DO DC24V/2A
Module type:	Output
F_Monitoring time:	120
CRC parameter:	4698
F_Source_Address:	2000
F_Target_Address:	198
PROFIsafe:	V2-MODE
Channel-granular passivation:	No

F-shared DB

Number of the F shared DB	DB818
Symbolic name:	F_GLOBDB
Collective signature address of the safety program:	
Absolute:	DB818.DBD 2
Symbolic:	"F_GLOBDB".F_PROG_SIG
Address for reading out the operating mode:	
Absolute:	DB818.DBX 36.0
Symbolic:	"F_GLOBDB".MODE
Address for reading out the error information:	
Absolute:	DB818.DBX 36.2
Symbolic:	"F_GLOBDB".ERROR
Address for reading out the compilation time:	
Absolute:	DB818.DBD 38

Symbolic:	"F_GLOBDB".F_PROG_DAT
Address for reading out RLO 0:	
Absolute:	DB818.DBX 36.3
Symbolic:	"F_GLOBDB".VKE0
Address for reading out RLO1:	
Absolute:	DB818.DBX 36.4
Symbolic:	"F_GLOBDB".VKE1

Supplementary information:

Safety mode can be deactivated:	Yes
Print created on:	10/13/2009 06:56:27 PM
Total pages printed:	5

OB1 - <offline>

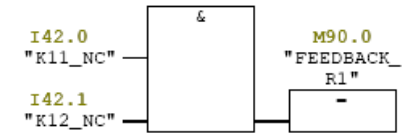
""

Name: **Family:**
Author: **Version:** 0.1
Block version: 2
Time stamp Code: 08/25/2009 05:43:34 PM
Interface: 02/15/1996 04:51:12 PM
Lengths (block/logic/data): 00262 00128 00020

Name	Data Type	Address	Comment
TEMP		0.0	
OB1_EV_CLASS	Byte	0.0	Bits 0-3 = 1 (Coming event), Bits 4-7 = 1 (Event class 1)
OB1_SCAN_1	Byte	1.0	1 (Cold restart scan 1 of OB 1), 3 (Scan 2-n of OB 1)
OB1_PRIORITY	Byte	2.0	Priority of OB Execution
OB1_OB_NUMBR	Byte	3.0	1 (Organization block 1, OB1)
OB1_RESERVED_1	Byte	4.0	Reserved for system
OB1_RESERVED_2	Byte	5.0	Reserved for system
OB1_PREV_CYCLE	Int	6.0	Cycle time of previous OB1 scan (milliseconds)
OB1_MIN_CYCLE	Int	8.0	Minimum cycle time of OB1 (milliseconds)
OB1_MAX_CYCLE	Int	10.0	Maximum cycle time of OB1 (milliseconds)
OB1_DATE_TIME	Date_And_Time	12.0	Date and time OB1 started

Block: OB1 "Main Program Sweep (Cycle)"

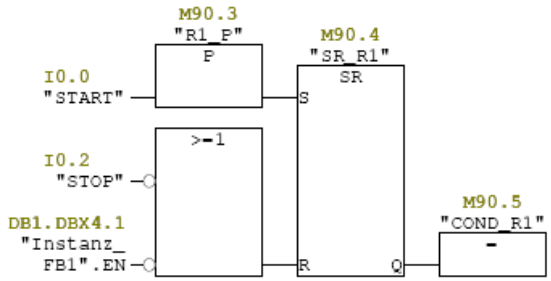
Network: 1



Symbol information

I42.0	K11_NC
I42.1	K12_NC
M90.0	FEEDBACK_R1

Network: 2

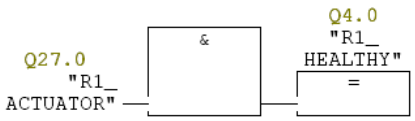


Symbol information

M90.3	R1_P
I0.0	START
I0.2	STOP
DB1.DBX4.1	"Instanz_FB1".EN
M90.4	SR_R1

M90.5 COND_R1

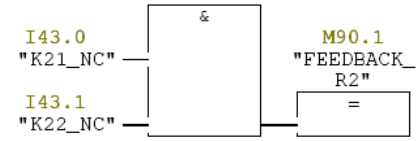
Network: 3



Symbol information

Q27.0	R1_ACTUATOR
Q4.0	R1_HEALTHY

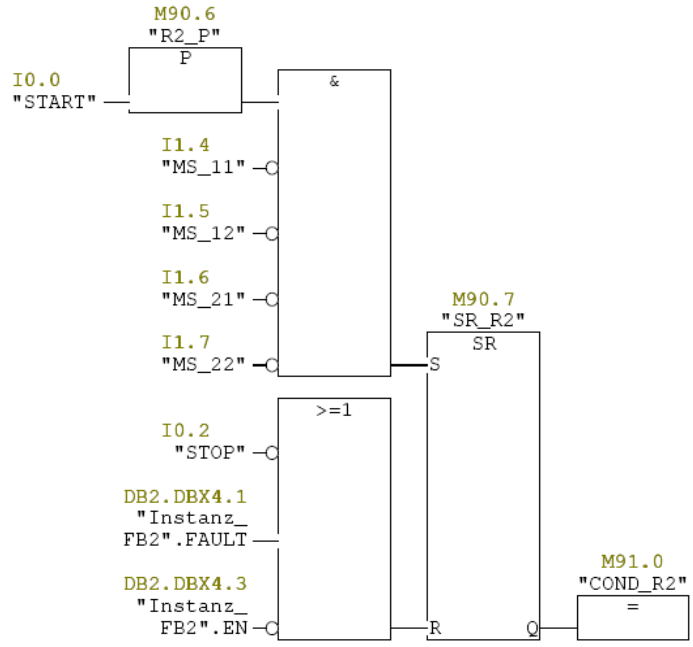
Network: 4



Symbol information

I43.0	K21_NC
I43.1	K22_NC
M90.1	FEEDBACK_R2

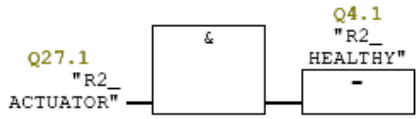
Network: 5



Symbol information

M90.6	R2_P
I0.0	START
I1.4	MS_11
I1.5	MS_12
I1.6	MS_21
I1.7	MS_22
I0.2	STOP
DB2.DBX4.1	"Instanz_FB2".FAULT
DB2.DBX4.3	"Instanz_FB2".EN
M90.7	SR_R2
M91.0	COND_R2

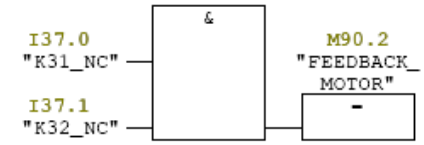
Network: 6



Symbol information

Q27.1	R2_ACTUATOR
Q4.1	R2_HEALTHY

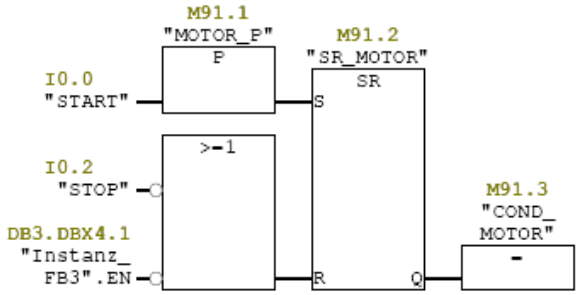
Network: 7



Symbol information

I37.0	K31_NC
I37.1	K32_NC
M90.2	FEEDBACK_MOTOR

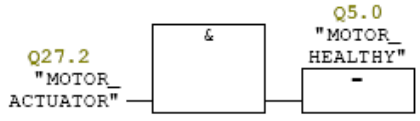
Network: 8



Symbol information

M91.1	MOTOR_P
I0.0	START
I0.2	STOP
DB3.DBX4.1	"Instanz_FB3".EN
M91.2	SR_MOTOR
M91.3	COND_MOTOR

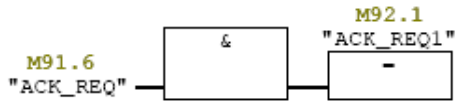
Network: 9



Symbol information

Q27.2	MOTOR_ACTUATOR
Q5.0	MOTOR_HEALTHY

Network: 10



Symbol information

M91.6	ACK_REQ
M92.1	ACK_REQ1

OB35 - <offline>

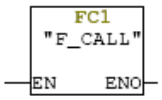
""

Name: **Family:**
Author: **Version:** 0.1
Block version: 2
Time stamp Code: 07/27/2009 03:16:26 PM
Interface: 02/15/1996 04:51:11 PM
Lengths (block/logic/data): 00130 00018 00022

Name	Data Type	Address	Comment
TEMP		0.0	
OB35_EV_CLASS	Byte	0.0	Bits 0-3 = 1 (Coming event), Bits 4-7 = 1 (Event class 1)
OB35_STRT_INF	Byte	1.0	16#36 (OB 35 has started)
OB35_PRIORITY	Byte	2.0	Priority of OB Execution
OB35_OB_NUMBR	Byte	3.0	35 (Organization block 35, OB35)
OB35_RESERVED_1	Byte	4.0	Reserved for system
OB35_RESERVED_2	Byte	5.0	Reserved for system
OB35_PHASE_OFFSET	Word	6.0	Phase offset (msec)
OB35_RESERVED_3	Int	8.0	Reserved for system
OB35_EXC_FREQ	Int	10.0	Frequency of execution (msec)
OB35_DATE_TIME	Date_And_Time	12.0	Date and time OB35 started

Block: OB35 "Cyclic Interrupt"

Network: 1



Symbol information
FC1 F_CALL

FC2 - <offline>

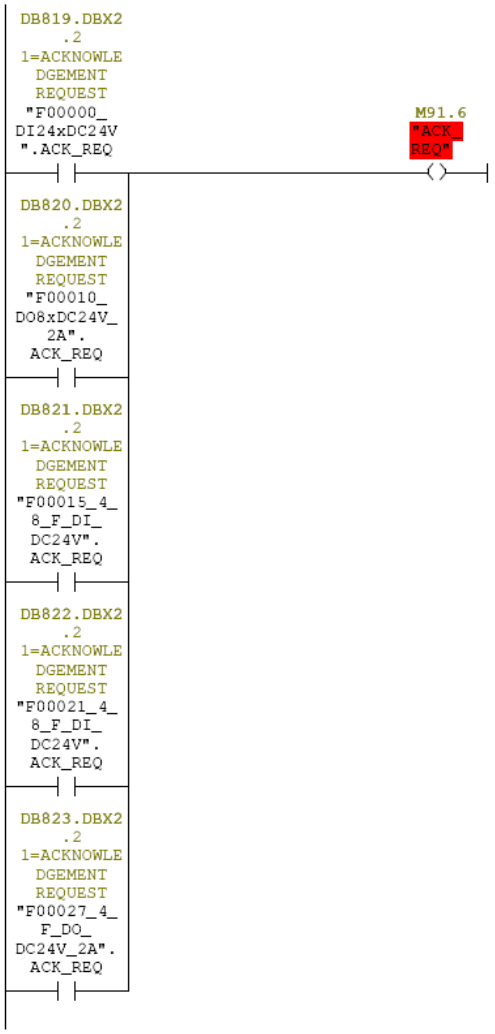
"F_Reintegration"

Name: **Family:**
Author: **Version:** 0.1
Block version: 2
Time stamp Code: 08/24/2009 06:37:22 PM
Interface: 07/27/2009 02:46:04 PM
Lengths (block/logic/data): 00186 00086 00000

Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC2

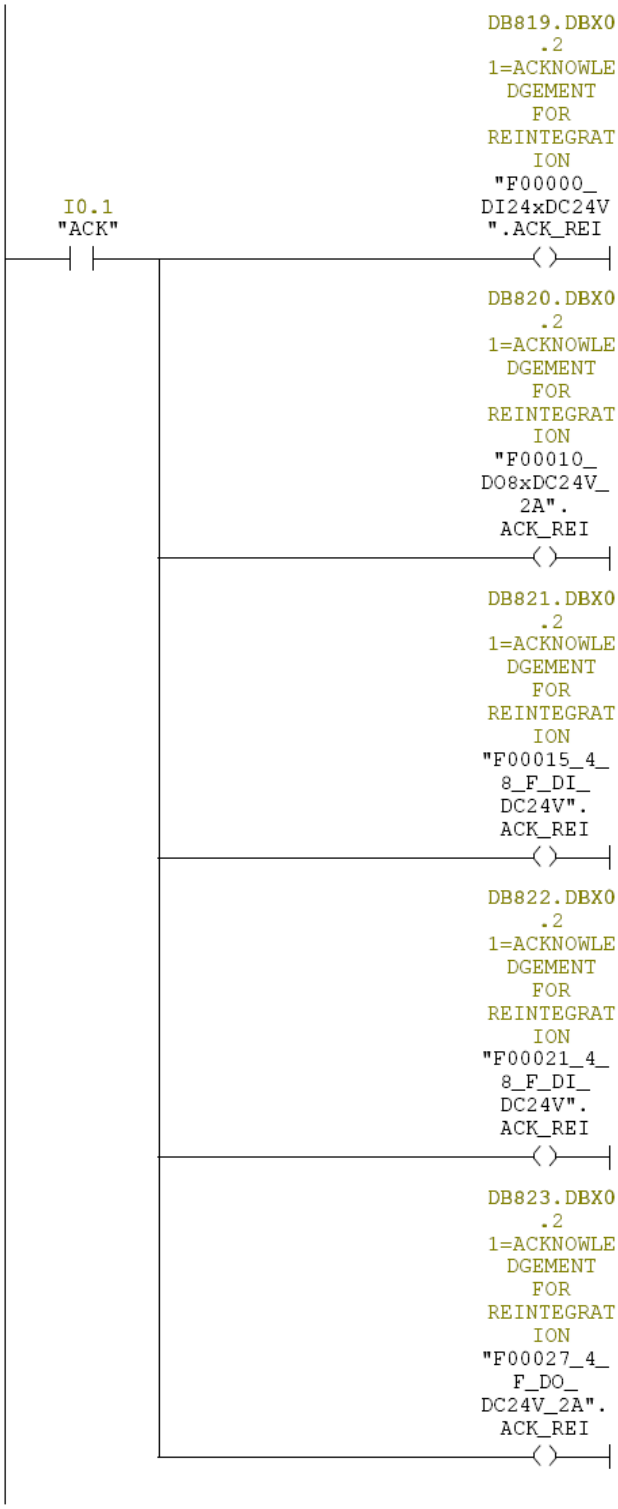
Network: 1 1=Acknowledgement necessary



Symbol information

```
DB819.DBX2.2 "F00000_DI24xDC24V".ACK_REQ       1=ACKNOWLEDGEMENT REQUEST
DB820.DBX2.2 "F00010_DO8xDC24V_2A".ACK_REQ     1=ACKNOWLEDGEMENT REQUEST
DB821.DBX2.2 "F00015_4_8_F_DI_DC24V".ACK_REQ   1=ACKNOWLEDGEMENT REQUEST
DB822.DBX2.2 "F00021_4_8_F_DI_DC24V".ACK_REQ   1=ACKNOWLEDGEMENT REQUEST
DB823.DBX2.2 "F00027_4_F_DO_DC24V_2A".ACK_REQ 1=ACKNOWLEDGEMENT REQUEST
M91.6        ACK_REQ
```

Network: 2 1=ACKNOWLEDGEMENT FOR REINTEGRATION



Symbol information

```
I0.1         ACK
DB819.DBX0.2 "F00000_DI24xDC24V".ACK_REI      1=ACKNOWLEDGEMENT FOR REINTEGRATION
```

```
DB820.DBX0.2 "F00010_DO8xDC24V_2A".ACK_REI    1=ACKNOWLEDGEMENT FOR REINTEGRATION
DB821.DBX0.2 "F00015_4_8_F_DI_DC24V".ACK_REI  1=ACKNOWLEDGEMENT FOR REINTEGRATION
DB822.DBX0.2 "F00021_4_8_F_DI_DC24V".ACK_REI  1=ACKNOWLEDGEMENT FOR REINTEGRATION
DB823.DBX0.2 "F00027_4_F_DO_DC24V_2A".ACK_REI 1=ACKNOWLEDGEMENT FOR REINTEGRATION
```

FC10 - <offline>

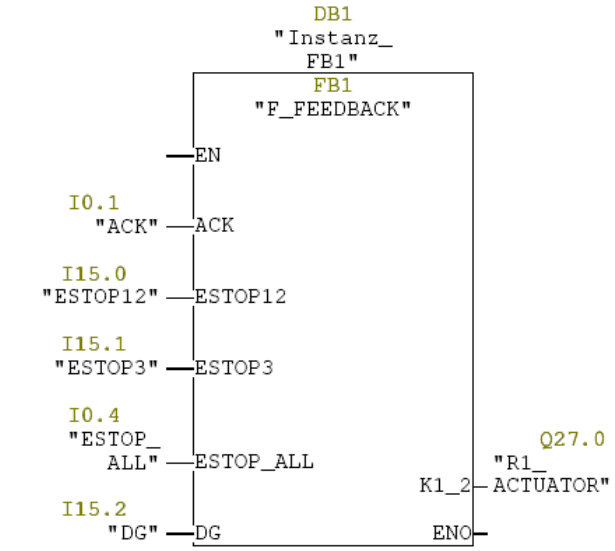
"Safety_Prgm"

Name: **Family:**
Author: **Version:** 0.1
Block version: 2
Time stamp Code: 08/25/2009 05:43:47 PM
Interface: 07/27/2009 02:53:48 PM
Lengths (block/logic/data): 00600 00490 00008

Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RELEASE	Bool	0.0	
K21_K22	Bool	0.1	
RETURN		0.0	
RET_VAL		0.0	

Block: FC10

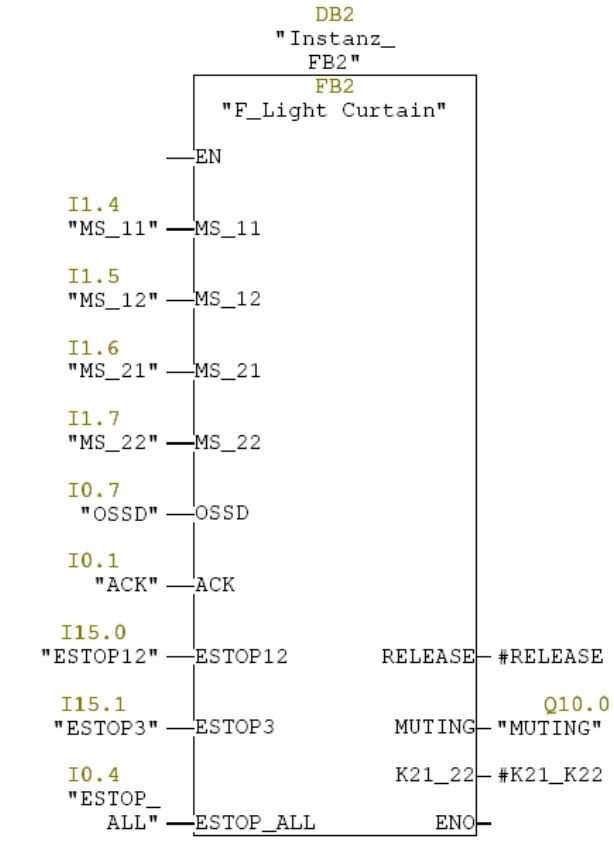
Network: 1



Symbol information

FB1	F_FEEDBACK
DB1	Instanz_FB1
I0.1	ACK
I15.0	ESTOP12
I15.1	ESTOP3
I0.4	ESTOP_ALL
I15.2	DG
Q27.0	R1_ACTUATOR

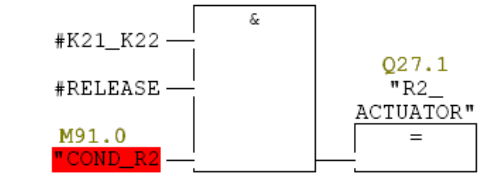
Network: 2



Symbol information

FB2	F_Light Curtain
DB2	Instanz_FB2
I1.4	MS_11
I1.5	MS_12
I1.6	MS_21
I1.7	MS_22
I0.7	OSSD
I0.1	ACK
I15.0	ESTOP12
I15.1	ESTOP3
I0.4	ESTOP_ALL
Q10.0	MUTING

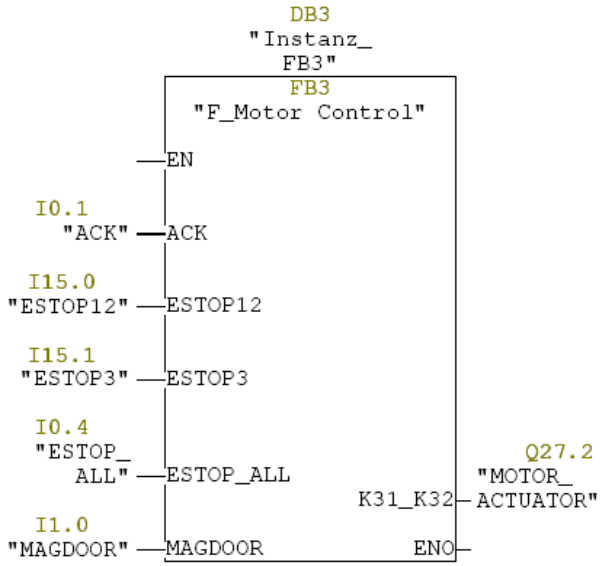
Network: 3



Symbol information

M91.0	COND_R2
Q27.1	R2_ACTUATOR

Network: 4



Symbol information

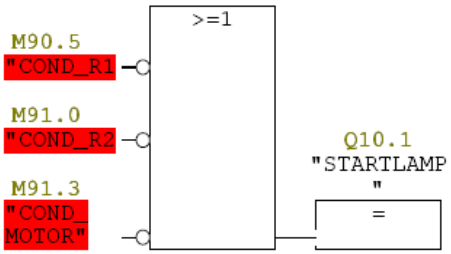
FB3	F_Motor Control
DB3	Instanz_FB3
I0.1	ACK
I15.0	ESTOP12
I15.1	ESTOP3
I0.4	ESTOP_ALL
I1.0	MAGDOOR
Q27.2	MOTOR_ACTUATOR

Network: 5

Symbol information

M92.1	ACK_REQ1
Q10.2	ACKLAMP

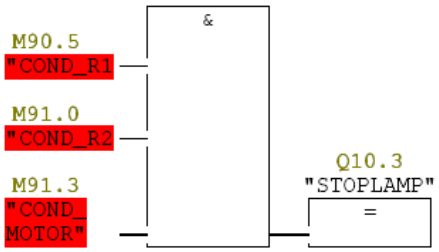
Network: 6



Symbol information

M90.5	COND_R1
M91.0	COND_R2
M91.3	COND_MOTOR
Q10.1	STARTLAMP

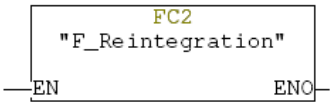
Network: 7



Symbol information

M90.5	COND_R1
M91.0	COND_R2
M91.3	COND_MOTOR
Q10.3	STOPLAMP

Network: 8



Symbol information

FC2	F_Reintegration

FB1 - <offline>

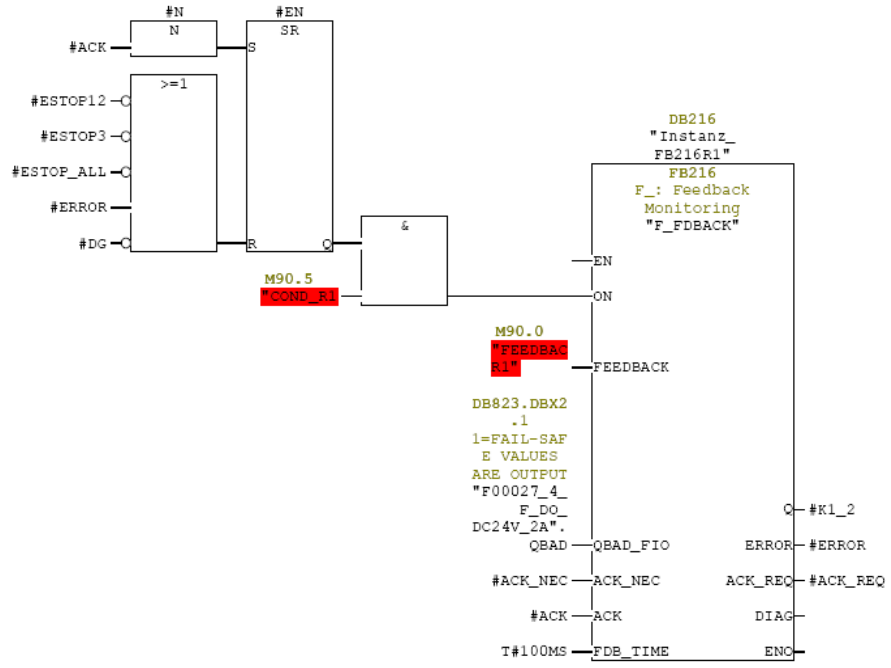
"F_FEEDBACK"

Name: **Family:**
Author: **Version:** 0.1
Block version: 2
Time stamp Code: 08/24/2009 06:46:23 PM
Interface: 08/24/2009 05:34:35 PM
Lengths (block/logic/data): 00326 00208 00006

Name	Data Type	Address	Initial Value	Comment
IN		0.0		
ACK	Bool	0.0	FALSE	
ESTOP12	Bool	0.1	FALSE	
ESTOP3	Bool	0.2	FALSE	
ESTOP_ALL	Bool	0.3	FALSE	
DG	Bool	0.4	FALSE	
OUT		0.0		
K1_2	Bool	2.0	FALSE	
IN_OUT		0.0		
STAT		0.0		
N	Bool	4.0	FALSE	
EN	Bool	4.1	FALSE	
ACK_REQ	Bool	4.2	FALSE	
ACK_NEC	Bool	4.3	FALSE	
ERROR	Bool	4.4	FALSE	
TEMP		0.0		

Block: FB1

Network: 1



Symbol information

```
M90.5         COND_R1
FB216         F_FDBACK                        F_: Feedback Monitoring
DB216         Instanz_FB216R1
M90.0         FEEDBACK_R1
DB823.DBX2.1 "F00027_4_F_DO_DC24V_2A".QBAD 1=FAIL-SAFE VALUES ARE OUTPUT
```

FB2 - <offline>

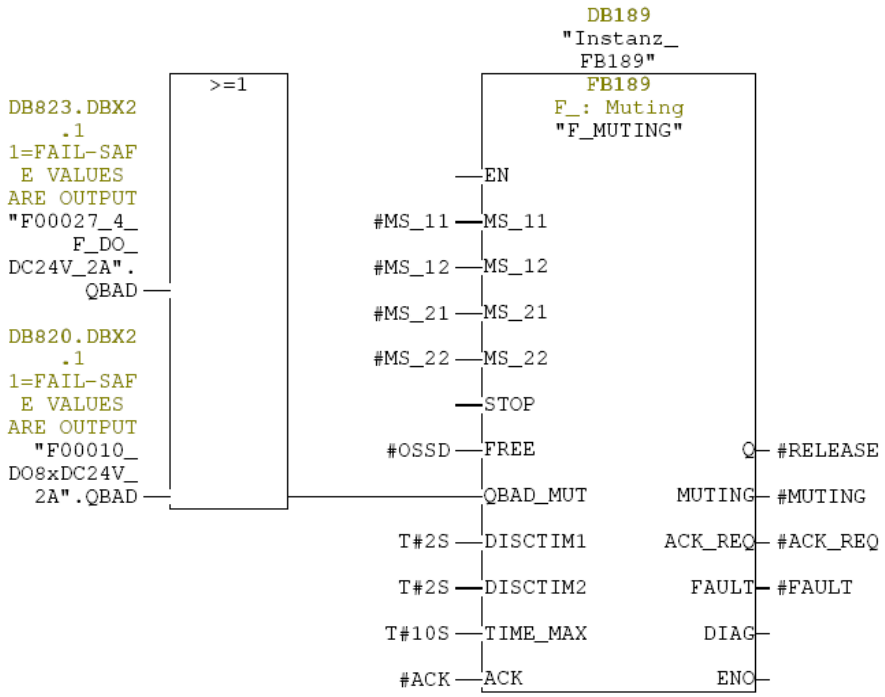
"F_Light Curtain"

Name: **Family:**
Author: **Version:** 0.1
Block version: 2
Time stamp Code: 08/24/2009 03:09:12 PM
Interface: 08/24/2009 03:09:12 PM
Lengths (block/logic/data): 00572 00442 00006

Name	Data Type	Address	Initial Value	Comment
IN		0.0		
MS_11	Bool	0.0	FALSE	
MS_12	Bool	0.1	FALSE	
MS_21	Bool	0.2	FALSE	
MS_22	Bool	0.3	FALSE	
OSSD	Bool	0.4	FALSE	
ACK	Bool	0.5	FALSE	
ESTOP12	Bool	0.6	FALSE	
ESTOP3	Bool	0.7	FALSE	
ESTOP_ALL	Bool	1.0	FALSE	
OUT		0.0		
RELEASE	Bool	2.0	FALSE	
MUTING	Bool	2.1	FALSE	
K21_22	Bool	2.2	FALSE	
IN_OUT		0.0		
STAT		0.0		
ACK_REQ	Bool	4.0	FALSE	ACK_REQ
FAULT	Bool	4.1	FALSE	
N	Bool	4.2	FALSE	
EN	Bool	4.3	FALSE	
ACK_NEC	Bool	4.4	FALSE	
ERROR	Bool	4.5	FALSE	
TEMP		0.0		

Block: FB2

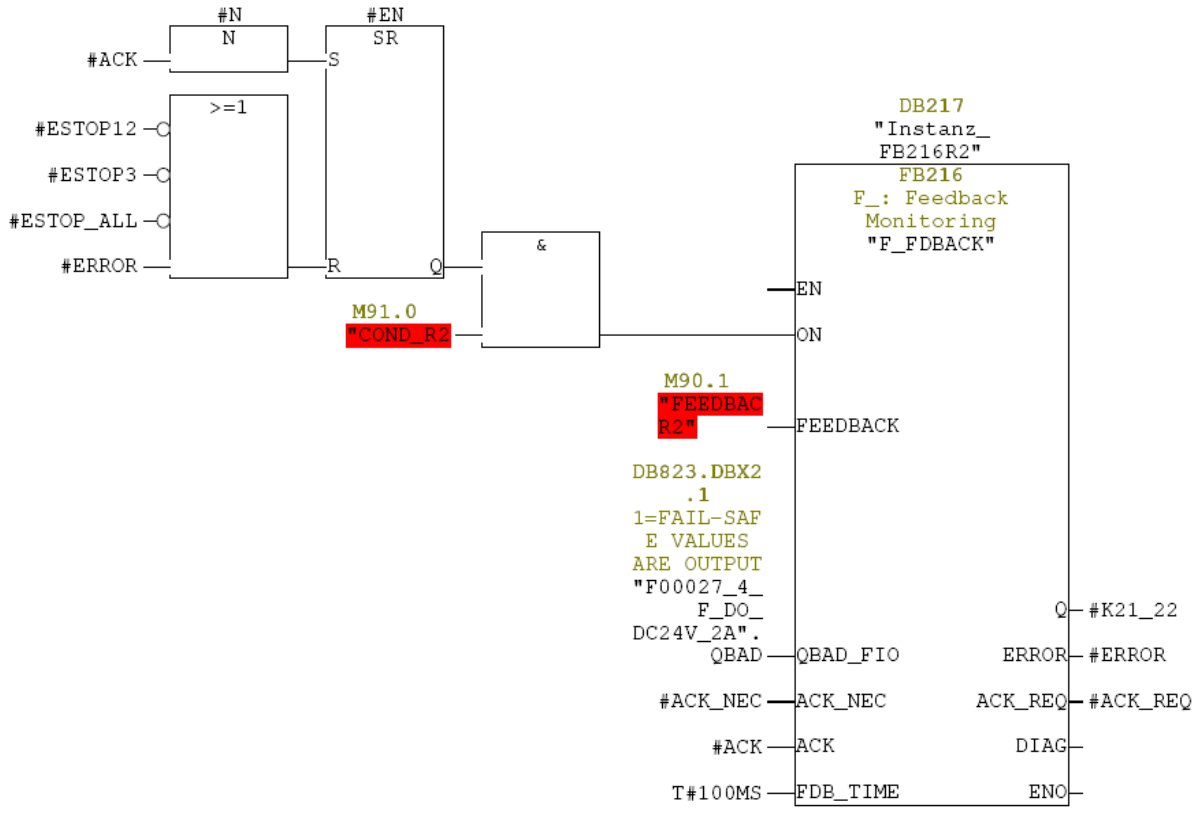
Network: 1



Symbol information

```
DB823.DBX2.1 "F00027_4_F_DO_DC24V_2A".QBAD 1=FAIL-SAFE VALUES ARE OUTPUT
DB820.DBX2.1 "F00010_DO8xDC24V_2A".QBAD    1=FAIL-SAFE VALUES ARE OUTPUT
FB189        F_MUTING                      F_: Muting
DB189        Instanz_FB189
```

Network: 2



Symbol information

```
M91.0         COND_R2
FB216         F_FDBACK                      F_: Feedback Monitoring
DB217         Instanz_FB216R2
M90.1         FEEDBACK_R2
DB823.DBX2.1 "F00027_4_F_DO_DC24V_2A".QBAD 1=FAIL-SAFE VALUES ARE OUTPUT
```

FB3 - <offline>

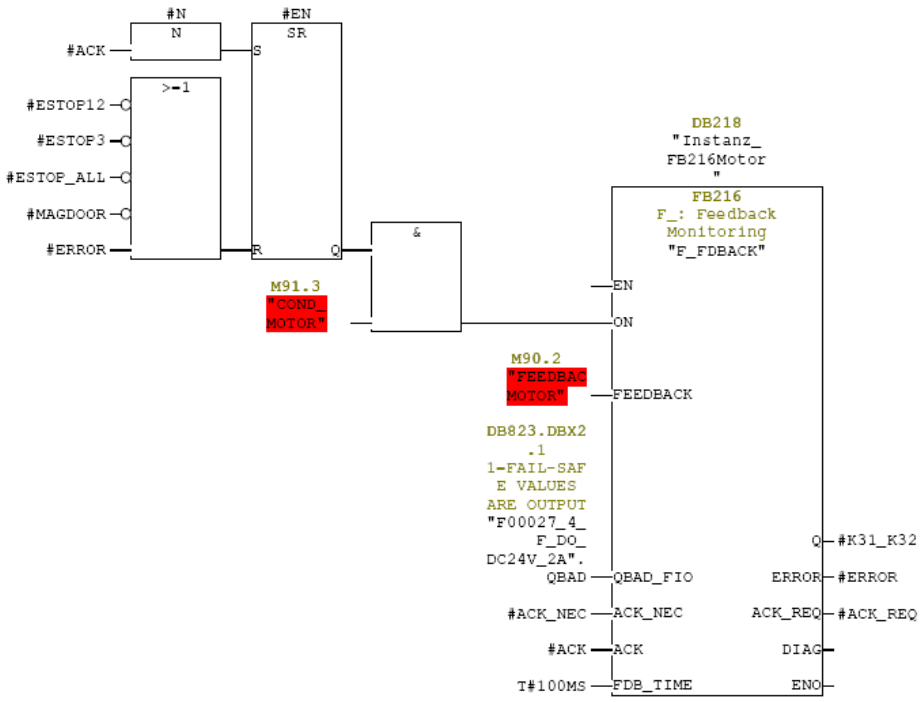
"F_Motor Control"

Name: **Family:**
Author: **Version:** 0.1
Block version: 2
Time stamp Code: 08/24/2009 05:53:58 PM
Interface: 08/24/2009 05:53:58 PM
Lengths (block/logic/data): 00322 00208 00006

Name	Data Type	Address	Initial Value	Comment
IN		0.0		
ACK	Bool	0.0	FALSE	
ESTOP12	Bool	0.1	FALSE	
ESTOP3	Bool	0.2	FALSE	
ESTOP_ALL	Bool	0.3	FALSE	
MAGDOOR	Bool	0.4	FALSE	
OUT		0.0		
K31_K32	Bool	2.0	FALSE	
IN_OUT		0.0		
STAT		0.0		
N	Bool	4.0	FALSE	
EN	Bool	4.1	FALSE	
ACK_REQ	Bool	4.2	FALSE	
ACK_NEC	Bool	4.3	FALSE	
ERROR	Bool	4.4	FALSE	
TEMP		0.0		

Block: FB3

Network: 1



Symbol information

```
M91.3         COND_MOTOR
FB216         F_FDBACK                        F_: Feedback Monitoring
DB218         Instanz_FB216Motor
M90.2         FEEDBACK_MOTOR
DB823.DBX2.1 "F00027_4_F_DO_DC24V_2A".QBAD 1=FAIL-SAFE VALUES ARE OUTPUT
```

Properties of symbol table

Name:	Symbols
Author:	
Comment:	
Created on	08/25/2009 05:25:09 PM
Last modified on:	08/27/2009 11:04:29 AM
Last filter criterion:	All Symbols
Number of symbols:	78/78
Last Sorting:	Symbol Ascending

Status	Symbol	Address	Data type	Comment
	ACK	I 0.1	BOOL	
	ACK_REQ	M 91.6	BOOL	
	ACK_REQ1	M 92.1	BOOL	
	ACKLAMP	Q 10.2	BOOL	
	COND_MOTOR	M 91.3	BOOL	
	COND_R1	M 90.5	BOOL	
	COND_R2	M 91.0	BOOL	
	DG	I 15.2	BOOL	
	ESTOP_ALL	I 0.4	BOOL	
	ESTOP12	I 15.0	BOOL	
	ESTOP3	I 15.1	BOOL	
	F_CALL	FC 1	FC 1	
	F_CTRL_1	FB 1639	FB 1639	
	F_CTRL_2	FB 1640	FB 1640	F_: Test Block an Programm Run Control
	F_DIAG_N	FB 1643	FB 1643	F_: Diagnosticbuffer Message with non CPU-Stop
	F_FDBACK	FB 216	FB 216	F_: Feedback Monitoring
	F_FEEDBACK	FB 1	FB 1	
	F_GLOBDB	DB 818	DB 818	
	F_IO_CGP	FB 1638	FB 1638	F_: Driver Block In-Output with Channel Granular Passivation
	F_Light Curtain	FB 2	FB 2	
	F_Motor Control	FB 3	FB 3	
	F_MUTING	FB 189	FB 189	F_: Muting
	F_Reintegration	FC 2	FC 2	
	F_TOF	FB 186	FB 186	F_: Timer Switch Off Delay
	F00000_DI24xDC24V	DB 819	FB 1638	
	F00010_DO8xDC24V_2A	DB 820	FB 1638	
	F00015_4_8_F_DI_DC24V	DB 821	FB 1638	
	F00021_4_8_F_DI_DC24V	DB 822	FB 1638	
	F00027_4_F_DO_DC24V_2A	DB 823	FB 1638	
	FEEDBACK_MOTOR	M 90.2	BOOL	
	FEEDBACK_R1	M 90.0	BOOL	
	FEEDBACK_R2	M 90.1	BOOL	
	FIMUTING	FB 1642	FB 1642	FI: Muting
	FITOF	FB 1641	FB 1641	FI: Timer Switch Off Delay
	HW_INT0	OB 40	OB 40	Hardware Interrupt 0
	I/O_FAULT1	OB 82	OB 82	
	I/O_FAULT2	OB 83	OB 83	
	Instanz_FB1	DB 1	FB 1	
	Instanz_FB189	DB 189	FB 189	
	Instanz_FB2	DB 2	FB 2	
	Instanz_FB216Motor	DB 218	FB 216	
	Instanz_FB216R1	DB 216	FB 216	
	Instanz_FB216R2	DB 217	FB 216	
	Instanz_FB3	DB 3	FB 3	
	K11_NC	I 42.0	BOOL	
	K12_NC	I 42.1	BOOL	
	K21_K22	M 91.5	BOOL	
	K21_NC	I 43.0	BOOL	

Status	Symbol	Address	Data type	Comment
	K22_NC	I 43.1	BOOL	
	K31_NC	I 37.0	BOOL	
	K32_NC	I 37.1	BOOL	
	MAGDOOR	I 1.0	BOOL	
	MOD_ERROR	OB 122	OB 122	
	MOTOR_ACTUATOR	Q 27.2	BOOL	
	MOTOR_HEALTHY	Q 5.0	BOOL	
	MOTOR_P	M 91.1	BOOL	
	MS_11	I 1.4	BOOL	
	MS_12	I 1.5	BOOL	
	MS_21	I 1.6	BOOL	
	MS_22	I 1.7	BOOL	
	MUTING	Q 10.0	BOOL	
	OSSD	I 0.7	BOOL	
	R1_ACTUATOR	Q 27.0	BOOL	
	R1_HEALTHY	Q 4.0	BOOL	
	R1_P	M 90.3	BOOL	
	R2_ACTUATOR	Q 27.1	BOOL	
	R2_HEALTHY	Q 4.1	BOOL	
	R2_P	M 90.6	BOOL	
	RACK_FAULT	OB 86	OB 86	
	RELEASE	M 91.4	BOOL	
	Safety_Prgm	FC 10	FC 10	
	SR_MOTOR	M 91.2	BOOL	
	SR_R1	M 90.4	BOOL	
	SR_R2	M 90.7	BOOL	
	START	I 0.0	BOOL	
	STARTLAMP	Q 10.1	BOOL	
	STOP	I 0.2	BOOL	
	STOPLAMP	Q 10.3	BOOL	

References

[1]: Toola, Arjan. "The safety of process automation." Automatica 29 (1993): 541-548.

[2]: Shen, K. C. "On the exploratory study of reliability and safety engineering techniques and their implementation in the machine design process." Reliability and Safety Engineering (1986).

[3]: Kletz, Trevor. Process Plant: A Handbook for Inherent Safety Design: 11.

[4]: http://www.osha.gov/

[5]: http://www.automation.siemens.com/cd/safety/html_76/produkte/si_ueberblick.htm

[6]: Summers, Angela E. "Techniques for assigning a target safety integiry level." ISA Transactions 37 (1998): 95-104.

[7]: MacDonald, David M. Practical Machinery Safety . Oxford: IDC Technologies, 2004.

[8]: Leighton, C. L. "Assessing the safety of existing plants- a case study." Reliability proceedings 89.1 (1989): 14-16.

*[9]: Bobbio, Andrea, et al. "Comparison of methodologies for the safety and dependability assessment of an industrial programmable logic controller." *.*

[10]: Daniels, B. K., and R. I. Wright. "Safety integrity assessment of programmable systems in UK industry." Compsac 84 proceedings (Nov. 1984): 440-452.

[11]: Wells, G. L. Safety in process plant design. London: George Godwin Limited, 1980.

[12]: Khan, Faisal I., and S. A. Abbasi. "Techniques and methodologies for risk analysis in chemical process industries." Journal of Loss Prevention in the Process Industries 11 (1998): 261-277.

[13]: Rausand, Marvin, and Knut Oien. "The basic of concepts of failure analysis." Reliability Engineering and System Safety 53 (1996): 73-83.

[14]: Redmill, F., M. F. Chudleigh, and J. R. Catmur. "Principles underlying a guideline for applying HAZOP to programmable electronic systems." Reliability Engineering and System Safety 55 (1997): 283-293..

[15]: Verna, Alfredo, and Geoff Stevens. "Is HAZOP always the method of choice for identification of major process plant hazards." IChemE (2008).

[16]: Walczak, Thomas. "Manufacturing purpose specific PLC based safety systems; 'Are all the bases covered?.'" ISA TECH/ EXPO 1.4 (1997): 161-168.

[17]: Maggioli, V. J., and E. I. Du Pont De Nemours & Company Inc. "Safety and programmable controller." Control Engineering Conference III (1984).

[18]: Douglass, and Bruce Powel. "Safety Critical systems design." Electronic Engineering 70.862 (98): 45.

[19]: Kanamaru, Hiroo, Tsuyoshi Mogi, and Naoki Aoyama. "Functional safety application using safety PLC." SICE Annual Conference, Kagawa University, Japan (Sept. 2007): 17-20.

[20]: Gall, Heinz, and Gerd Rabe. "International and European standardization for PLCs in safety critical systems- Qualification, type testing, certification and licensing ." ISA Transactions 34 (1995): 273-281.

[21]: "Functional safety in process instrumentation with SIL rating- Questions, examples, background." Siemens AG 2007 (2007).

[22]: Krosigk, Hartmut. "Functional safety in the field on industrial automation- The influence of IEC 61508 on the improvement of safety-related control system." Computing and control engineering (Feb. 2000).

[23]: "Functional Safety and IEC 61508." IEC (Sept. 2005).

[24]: "Functional safety of electrical/ electronic/ programmable electronic safety-related systems." IEC.

[25]: Manfred, Kraemer. "Failsafe PLC tackles critical applications." Engineering & Automation XIII.1(1991).

[26]: Goble, W. M. "Using PLCs in safety applications." Hydrocarbon Processing, Pennsylvania (June 1996).

[27]: Hoske, Mark T. "Back to basics- safety PLCs." Control Engineering (Aug. 2005).

[28]: "Safety integrated technology forges ahead at Opel body shop safety at Body shop." Siemens move up safety special, case studies. May 2004.

[29]: TUV ASI. 11 Feb. 2009 <http://www.tuvsai.com/>.

[30]: "IEC 61508 Standard part 1 to 7, Functional safety of electrical/ electronic/ programmable electronic safety-related systems", First Edition (1998-12)

[31]: Rockwell Automation, "Safety Relay vs. Safety PLC: Choosing the Right Safety Control Architecture", White Paper, Publication SAFETY-WP001A-EN-E — March 2002

[32]: S. Kmenta and K. Ishii, "ME317 dfM: Product Definition Failures Modes and Effects Analysis" (January 2003).